Digital Forensic Evidence Examination

by Fred Cohen, Ph.D.

5th Edition

Digital Forensic Evidence Examination

Table of Contents

Front matter

Digital Forensic Evidence Examination - 4th Edition

is

ISBN # 1-878109-48-0

Published by Fred Cohen & Associates out of Livermore, CA.

2013-06-01

We are not liable for anything you do with this book or use this book for, including but not limited to choking people with its pages, using the book to trip folks, the kid leaving it on the steps, the pollution it creates when you burn it, or any side effects you may have in the afterlife from having sworn about certain things I may have said. All of these things as well as any other thing that happens as either a direct or indirect consequence of my having written this book is not my fault, and I take no responsibility and MAKE NO WARRANTY EITHER EXPRESSED OR IMPLIED ABOUT ANYTHING IN THIS BOOK. The law says I have to use large block letters to say that so you are amply warned.

If you are offended by anything I have written, or if you feel your children or parents have been corrupted by it, remember, this warning was given to you and you could have looked at it before reading the book or buying it. Or lacking that, you could have returned it, sold it to someone else, or given it to someone you don't like.

WARNING! READ THIS PAGE BEFORE BUYING THE BOOK!

That should do it I think...

Forward

Welcome to Digital Forensic Evidence Examination.

This is a science book designed for advanced graduate students working on their Ph.D. in digital forensics. As such, it is not easy reading, it doesn't have a lot of simple examples, it has symbols that look like mathematics, and it talks about the limits of what can and cannot be done under different assumptions that may or may not be the case.

If you are looking for a book telling you how to make a forensically sound copy of a disk drive and run it through a commercial program to search for some string, look no further. You are looking in the wrong place and this is not the book for you.

If you are looking for advanced understanding that requires substantial thought and effort on your part, and if you have a reasonably strong background in the issues of digital forensics, a computer engineering or computer science background, a legal background, or other similar background at the Masters level or beyond, then this is the book for you.

This book is designed as an advanced graduate text for professors and students seeking to gain in-depth knowledge of the foundation and underpinnings of digital forensic evidence examination and put their understanding of the field into a more scientific context. It is also well suited for use by professional digital forensic evidence examiners who want to expand their understanding of the field they work in. This is not a book of techniques or tricks, nor is it intended to be a practical guide to doing the day-to-day work that most DFE examiners do today. It is a book intended to help define a science.

This is the 4th edition of this book, and this book is increasingly becoming a stable document reflecting a stable basis with relatively little change over time. That is a good sign for the development of digital forensic evidence examination as a science.

It is the hope of this book to grow with the field and continue to trace the theoretical aspects of the science of DFE examination along with practical implications of that basis.

Appreciation

This book was originally written as the stock market crashed, jobless rates were increasing, my wife was starting her dissertation, one of my children was about to graduate from college and two others were in college, consulting work was hard to come by, and California Sciences Institute was just getting started. The 2nd edition was put in place a year later, the 3rd at the end of the worst economic year for me in the last ten, and this 4th edition was started as the last one was sent to press.

By now, all of my children are either in college or graduate school, my wife is pursuing her career as a psychologist, and CalSci is a licensed educational institution trying to grow and thrive.

Many folks have now patiently and repeatedly copy edited and reviewed the versions of this books, and I will likely miss many of them. But again, I want to thank Charles Preston and Pete Shehu who each carve out days and days from their lives to read, reread, and comment on these and other works. Other peer reviewers who have substantially contributed include, Betsy Nichols, Ovie Carroll, Chet Uber, Gary Kessler, Julie Lowrie, and the editorial boards and reviewers of the IFIP and IEEE conferences, encyclopedias, books, journals, and other venues who have peer reviewed papers that form content that ends up part of the book. The Diplomatics and archival science portion of the book was based on work, reviewed, and corrected by Luciana Duranti. Professors and their graduate students have also reviewed and commented, members of several professional societies have weighed in, some of the material has been published as articles in peer-reviewed journals, and some of the work is increasingly finding its way into other works. Conference participants too numerous to name have also reviewed portions of this book, sometimes in real-time, and commented on issues and improvements. In many cases they led me to updates as I add their work to the body of knowledge covered.

The book is increasingly being used as part of graduate programs around the World, and as a result, we are getting more discussions with graduate students about potential research areas, subjects for dissertations, and other related matter. This is a true sign of progress, and my appreciation goes out to you all.

About the Author

Dr. Cohen earned his B.S. in Electrical Engineering from Carnegie-Mellon University in 1977, M.S. in Information Science from the University of Pittsburgh in 1980, and Ph.D. in Electrical Engineering from the University of Southern California in 1986, focussed on computer engineering and with a dissertation titled "Computer Viruses", a term that he first defined in 1983.

He has worked professionally with computers since the early 1970s and has been involved in legal matters related to digital forensics since the middle of the 1980s. He has taught courses and students from government agencies like the United States Secret Service, the National Security Agency, branches of the US Department of Defense, State police from many states, agencies that are now members of the Department of Homeland Security, and other intelligence agencies. For several years, he led a research and development team at Sandia National Laboratories that developed digital forensics methods and mechanisms, many of which are in use by government agencies and other organizations, participated in the development of national guidelines for, and continues to research and develop tools and methods for digital forensics.

Dr. Cohen worked as a research professor creating and teaching graduate level courses in related areas for the University of New Haven, collaborated in the creation of new curriculum for doctorate level graduate degrees in digital forensics for the California Sciences Institute, taught as a guest instructor in digital forensics for the Federal Law Enforcement Training Center, acted as a California POST certified law enforcement instructor in digital forensics, participated in the New York Electronic Crimes Task Force before there were regional task forces, and currently participates in the San Francisco Electronic Crimes Task Force, both run by the United States Secret Service. He is also a founding Digital Forensics Certified Practitioner.

He has published more than 250 articles and professional works, is a member of editorial boards and peer reviews papers for professional publications, authored chapters in books and two full books on this subject, and continues to be active in research, development, and application of digital forensics methodologies, tools, and techniques. In 2011 he was named as a fellow of the ISC^2 and is thus now a life member of ISC^2, and was awarded an honorary Ph.D. (*honoris causa*) by the University of Pretoria for his work in this and related areas.

Dr. Cohen is currently President of California Sciences Institute, where he leads the Ph.D. program in digital forensics, and CEO of Fred Cohen & Associates, a firm that does expert witness work including digital forensic examination, DARPA funded research using and advancing forensic techniques, and consulting in information protection. He continues to do research, develop software and systems, teach, write and peer reviewed articles, attend and speak at conferences and in other venues, participate in professional forums and on editorial boards, and write short articles on digital forensics and related issues. Many of his works are available for free to all through the all.net Web site.

1 Introduction and overview

This book is about the science of examining digital forensic evidence in and for legal settings and purposes.

Background

Like almost every scientific endeavor, the examination of digital forensic evidence (DFE) started out somewhere between an art and a craft. People with special skills and knowledge leverage that skill set and knowledge base to put forth notions about the meaning of DFE in the context of legal matters. While the court system greatly appreciates science and its role through expert testimony in providing probative information, that appreciation is substantially challenged by the lack of a scientific base, in the form of peer reviewed publications associated with professional societies, a well-defined and well understood body of knowledge, an underlying scientific methodology that the courts can understand, an experimental basis, and all of the other things that go with normal science. As the volume and criticality of DFE has increased, there has been an increasing recognition of the limitations of DFE, and more importantly, the limitations of the underlying science and its proper application in legal settings.

In making progress in the science of digital forensic evidence examination, it may be helpful to look at the advancement of science in other areas. In most areas of science, a scientific methodology consists of four basic elements; (1) studying the past as well as current theories, methods, and experimental bases; (2) identifying inconsistencies between current theories and repeatable experimental outcomes, (3) hypothesizing new theory that explains refuted hypotheses, and performing experiments to test the new theory, and (4) publishing the results. However, in an area where there is no pre-existing scientific theory, a new epistemology, methodology, theory, experimental basis, and perhaps even a new physics has to be built from scratch. In the case of DFE examination, this book represents one attempt to fuse the limited historical areas of theory and practice in the relevant sciences and engineering disciplines into a new overarching scientific view of DFE examination.

The call for a science

The US Supreme Court has spoken[1] and the National Research Council has concurred.[2] A rigorous scientific approach is needed for forensic evidence to warrant use in the courts in the United States, and much of the world is likely to follow that approach, if it isn't already following it.

To a substantial extent, this call for science stems from failures of forensics. Recent failures have been quite dramatic. For example, in the Madrid bombing case, where the US FBI declared that a fingerprint from the scene demonstrated the presence of an Oregon attorney. However, that attorney, after having been arrested, was clearly demonstrated to have been on the other side of the world at the time to question. The side effect is that fingerprints are now being challenged as valid scientific evidence across the land, and around the World.[3] A similar situation exists in cases where forensic examiners have done a poor job and testified in numerous cases, typically for the prosecution. The inability to effectively challenge evidence by such supposed experts through a scientific methodology and inquiry process makes this sort of evidence extremely problematic, and all the more so because of the limits of human integrity. In case after case, when the details are examined, forensic evidence seems to come up short under close scrutiny, and if competently challenged. The solution is simple. Build and apply real science, and the truth will out.

An ongoing attempt at proposing a science

This ongoing attempt to propose a science for DFE examination consists of; (1) the ongoing search for historical areas that may be brought to bear; (2) the ongoing update and enumeration of some elements of an epistemology and physics of digital information; (3)

1 Daubert v. Merrell Dow Pharmaceuticals, Inc. 509 US 579, 125 L. Ed. 2d 469, 113 S. Ct. 2786 (1993).

2 Committee on Identifying the Needs of the Forensic Sciences Community, "Strengthening Forensic Science in the United States: A Path Forward", ISBN: 978-0-309-13130-8, 254 pages, (2009).; Committee on Applied and Theoretical Statistics, National Research Council.

3 Statement of Glenn A. Fine, Inspector General, U.S. Department of Justice before the House Committee on the Judiciary Subcommittee on Crime, Terrorism, and Homeland Security concerning “Section 1001 of the USA Patriot Act” May 10, 2005, at: http://www.usdoj.gov/oig/testimony/0505b.htm

a model of the DFE examination process within the context of the legal environment; (4) the interpretation of existing information, experimental results, and theory in the proposed model; and (5) the study of the state of consensus of this model in the scientific community. A brief overview of the situation as it stands as of this writing is included here.

Scientific disciplines of archival science and diplomatics

Legal systems over several millennia have had to deal with issues related to the admission and use of informational evidence in legal matters. This ranges from documents associating ownership of property through the emergence of fingerprints as evidence and their near demise. As an overarching science, the areas of archival science and diplomatics are among the oldest and most deeply embedded in the legal systems of the World, and are thus a good starting point.

Archival science started as a scientific body of knowledge at least in ancient Rome, were the records of government were written on wax tablets and transported through underground passageways to the central archives for permanent archival preservation. Such records were tracked and made available to the public only in certified copies produced by the archivists who were government employees trusted to diligently perform their duties (quaestores).[4]

The Justinian code codified the definition of archives as “the place where public records are deposited” … so that “they remain uncorrupted and might be found rapidly by those who request them”, and so they “preserve perpetual memory [of] the acts [to which they relate]”. These principles and approaches have been taught since 1158 in all of the legal educational systems associated with “common law” and formed the foundation for admissibility of records and reliance upon them. By the 1500s these ideas became a far more widespread subject of research and implementation, and various facets of understanding relating to the trustworthiness of records were studied and put in to practice over the centuries.

4 L. Duranti, “Archival Science”, Article in Encyclopedia of Library and Information Science.

Diplomatics background

In 1681, the archival science was codified into a legal framework[5] which focused on individual documents, their characteristics, genesis, and treatment.[6] Archival science and diplomatics were developed together, and in the 1800s laws were increasingly being formulated taking into account their concepts and methodologies. By the late 1800s, rules of evidence and their foundation were explained in detail and by the early 1900s, they were clearly codified into laws globally. Today, diplomatics is being updated and applied to information age records around the World for public and private archival organizations. It remains the basis for much of the legal system, and as such, forms a scientific basis for understanding digital evidence.[7]

"According to modern diplomatics, a record is a document created (i.e., made or received and set aside for action or reference) in the course of activity as an instrument and by-product of it."[8] The field of diplomatics focuses on the **assessment of the trustworthiness of records**, which is done retrospectively for existing records (and in digital forensics), and prospectively for designing record systems and types. Classic diplomatics associates **trustworthiness** with **authenticity of the records** (they were written at the time claimed and signed by a person competent to produce them). Modern diplomatics defines and assesses "**trustworthiness**" in terms of **reliability**, **authenticity**, **accuracy**, as a basis to **authenticate** a record.

Status of Transmission

There are different record states: "original" ("internal" or "external"), "draft", and "copy".

An "**original record**" is **the first record generated in a complete form capable of reaching the consequences intended by its author**. Even a forgery has an "original" state.

5 Dom Jean Mabillion, "De Re Diplomatica", 1681, Saint-Maur, France.
6 L. Duranti, "Diplomatics: New Uses for an Old Science", Archivaria 28. 7-27, 1989.
7 L. Duranti, "Diplomatics", Encyclopedia of Library and Information Sciences, Third Edition DOI: 10.1081/E-ELIS3-120043454, 2010, Taylor & Francis.
8 Ibid.

An original record has the properties of "**primitiveness**", "**completeness**", and "**effectiveness**".

For an "**external**" record **meant for transmission across space** (e.g., and email or a Web form), the "original" is the document received by the addressee, while "**internal**" records **intended only to be transmitted over time** (e.g., internal records in a filing cabinet), the original is the record kept by the author.

In the digital environment, the "**original**" is t**he first manifestation of the record, either received or stored, depending on whether the record is external or internal.** There are multiple original records when, for example, an email is saved at origin and sent to another party or parties, when there is a treaty or contract signed by multiple parties, each of which keeps a copy, and so forth.

A "**draft**" is a document **prepared for the purpose of correction, and it is meant to be provisional and temporary**. It may be at various levels of completion, but it is never an effective or legal document and, if kept, is intended to stay with the author. Thus it may not be transmitted across space, but only over time. If it is electronically circulated, each recipient then has an "original" of that draft with regard to status of transmission because it was transmitted to the intended recipient at the level of completion and was thus capable of achieving its purpose of being examined, stored, or used in some way by the recipient(s). It is a draft only as to content.

A "**copy**" is **a reproduction of another document**, which could be an "original", a "draft", or another "copy".

The most trustworthy copy is a **copy in the form of original**, which is **identical to the original in all respects**, including holographic signatures, if required, but is issued after the original. Equally trustworthy is the **authentic copy**, which is **declared to conform to the original by an official entrusted with such responsibility**.

An "**imitative copy**" (e.g., a photocopy) is **a reproduction of both the form and content of a record**, while a "**simple copy**" only **transcribes the record content**, and "**inserts**" are **original records containing a copy of another record or part of it**. In most, if not all computer situations, DFE examiners only see at best

a “copy in the form of original” because the original writing is the instantiation in the media as created, and is physical rather than digital in form. In computers, an imitative copy of the original is normally stored in a different media and the original does not persist. The copy made for examination is not made by the author.

Reliability: the record as a true statement of fact

Reliability of a record relates to the extent to which it reflects the reality it purports. It is assessed on the basis of (1) **completeness**, which is to say, the record has all of the formal elements required by the administrative or legal system for achieving its purpose; and (2) the **controls exercised in its creation**, such as the competence of the author in terms of the authority and capacity to issue the record. Reliability is the responsibility of the record's **creator**, (i.e., the person and/or organization that made or received the record and maintained it with its other records).

Authenticity: a record has not been tampered with or corrupted

An **authentic** record is one that (1) **preserves** the same **identity** it had when first generated and (2) can be presumed or proven to have **maintained** its **integrity over time**.

The **identity** of a record is made up of all the characteristics that distinguish it from any other record (e.g., name of its author and addressee, date of compilation, title, tracking codes, etc.). Identity is assessed based on the formal elements on the face of the record, and/or attributes, as expressed in redundant records. (e.g., metadata, registry in a repository, etc.)

The **integrity** of a record is associated with its ability to convey the (entire) message it was intended to communicate when generated. Whether the ink is fading, the media is falling apart, or the bits are not identical to the first manifestation of the record, if the content is readable, complete, and the same as was originally intended for all material purposes, integrity is considered maintained.

Integrity of a record is **inferred from** both **appearance**, which might be deceptive (e.g., a good forgery) **and** from the **circumstances of** its **maintenance and preservation**. The chain of responsible and legitimate custody is generally considered an insurance of integrity unless there is proof to the contrary.

The authenticity of a record is a responsibility that moves from party to party together with its legitimate custody, starting at the creator, to the various preservers, who must guarantee it over the life cycle of the record.

Accuracy: truthfulness, exactness, precision, or completeness

Accuracy is the **trustworthiness of the data** (i.e., the smallest meaningful indivisible piece of information) within a record. It is defined as **truthfulness**, **exactness**, or **precision**. Accuracy is usually presumed for reliable or authentic records.

The ease of corruption of digital information during transmission across space and over time makes accuracy a responsibility that also moves through time and space from party to party.

Authentication: declaration of authenticity made by competent party

Introduction of evidence into legal proceedings essentially always requires that the evidence be authenticated by a person or persons. The most trustworthy records are typically considered to be records that are declared authentic by someone with formal responsibility as custodian of those records. **Authentication** is a **declaration of authenticity** by a competent custodian or public official. It consists of a either a **sworn statement** or an **element added to the record after its completion**. An added element is typically something like a seal, stamp, or symbol. A digital signature may be such a seal.

Authentication is **not the same as authenticity**. Authenticity is a quality of the record (i.e., it is what it purports to be) as long as it exists. Authentication only guarantees that a record is authentic at one specific moment in time, when the declaration is made or the authenticating element (e.g., a seal, stamp, or symbol) is affixed.

Building the diplomatics discipline from the definitions

“The building blocks used by classic diplomatists were the **juridical system**, which is the context of records creation; the **act**, which is the reason for records creation; the **persons**, which are the agents; the **procedures**, which guide the actions and determine their documentary residue, the **documentary form**, which reflects the act and allows it to reach its purpose, and the **archival bond**, which reveals the relationship of a record with all the other records in the

same aggregation."[9] (emphasis added) We rely heavily on this paper herein and throughout this discussion.

Modern diplomatics has extended the concept beyond the judicial system, but for the purposes of this work, it is the judicial context that is of the most interest. The five modern contexts in which each record exists are: (1) **juridical/administrative**, which the name clarifies, (2) **provenencial**, which deals with the body creating the records (i.e., provenance), (3) **procedural**, which is the series of steps by which the record is created, (4) **documentary**, which is the body of records, the system of record keeping, its metadata, structures, organization, etc. in which the record resides, and (5) the **technological context**, which is the technology in which the record is created and resides.

Acts

In classical diplomatics, an **act** is an **exercise of will intended to produce effects**. **Transactions** are **acts aimed to create, modify, maintain, or extinguish relationships** between two or more physical or corporate persons. **Some acts**, especially transactions, **occur in writing**, thereby **resulting in records**.

> **Notitia** is a record that was meant to provide evidence of an act that came into existence and was complete before being manifested in writing. These are called **probative** records and are certificates, registrations, transcripts, and receipts.
>
> **Charta** is a record meant to put the act into being and is therefore the essence and substance of the act. These are called **dispositive** records and include contracts, grants, applications, and money orders.

For these records, a written form is required by the juridical / administrative system within which they are created. Thus they are **legal records**.

Nonlegal records can are distinguished into (1) **supporting records**, that support the activities they are part of (e.g., notes, maps, plans, etc.); (2) **narrative records**, that are free-form communications of information (e.g., emails, memos, and other

9 L. Duranti, "Diplomatics", Encyclopedia of Library and Information Sciences, Third Edition DOI: 10.1081/E-ELIS3-120043454, 2010, Taylor & Francis.

communications); (3) **instructive records**, that indicate the form in which things are to be presented (e.g., regulations, manuals, instructions for forms, scores, scripts, etc.); and (4) **enabling records**, that (a) **enable performance of mechanisms** (e.g., software, upgrades, etc.), (b) **execute business transactions** (e.g., a business application), (c) **conduct experiments** (e.g., mechanisms by which experiments were carried out), and (d) **analytical or observational data** (e.g., analytical programs, the data they store and process, etc.). Neither supporting nor narrative records provide evidence that any such act was actually carried out, while both instructive and enabling records are stored differently than they are viewed (i.e., the are “latent”)

Persons

Persons (as opposed to people) are the **subjects of rights and duties**. They are recognized by the legal system as **capable of acts**. The same entity may be a person or not for the purposes of different acts, even in the same legal matter. For example, women may be persons for the purpose of getting paid but not for the purpose of voting in some societies. Persons are **physical** (e.g., human beings) or **judicial** (e.g., a corporation, an estate of a dead person, or a succession of persons in a position or title).

For a record to come into existence, an **author**, **writer**, and **addressee** are necessary. The **author** is the **person with the competence** (i.e., authority and capacity) **to issue the record**, which is made in its name or by its order (e.g., the insurer that issues a payment, the university that appoints a professor). The **writer** is the **person competent for the articulation and disclosure of the record** (e.g., the broker who signs insurance checka or dean who signs appointment letters). The **addressee** is the **person for whom the record is intended**. For example, the Congress (author) may proclaim via law issued by the congressional printing office (the writer) that anyone who stakes a claim of 10 acres in a new territory will be granted a tax exemption. (the internal revenue service is the addressee).

Modern diplomatics also identifies the **creator**, a person among whose records (**archival fonds**) a record is filed. In digital systems, there is also the notion of the **originator** who is **responsible for**

the electronic account or space in which the record was generated or from which it is sent. For example, there may be a user identity (originator) associated with the person who transmitted the letter of appointment to the human resources department.

A **public record** is a record **issued by a public person**, (e.g., a driver's license issued by the department of motor vehicles) while a **private record** is a record **issued by a person "deprived" of public function** (e.g., a business record of a private company or a letter between friends). When there is a mix of public and private issuance (e.g., the IRS specifies the form and the taxpayer fills it out) the nature of the person dictating form and/or procedure determines if the record is private or public.

Procedure

A **procedure** is **a formal sequence of steps** by which a **transaction** is carried out. A **process** is a series of motions by which a person prepares to carry out acts, including acts involved in a procedure. Classic diplomatics identifies two types of procedures:

> **The procedure governing the act** can be **initiated** either **by the person carrying out the act** (i.e., a decision by the **iussio**) or **by another person** (i.e., (1) a petition to the authority or request to accomplish something followed by (2) the intercession or recommendation, (3) the intervention or permission of persons affected by the action or its consequences, (4) the **iussio**, or command to create the record enacting the act or evidencing it. After the procedure controlling the act is completed, the procedure producing the documentation begins.
>
> **The procedure governing documentation** of the act includes (1) the production of a **draft**, (2) preparation of a **fair copy**, (3) unabridged or abridged **registration**, (4) the **validation** through signatures or affixing of stamps, seals, etc., (5) computation of **taxes** for the issuing of the record, and (6) **delivery** of the record.

Modern diplomatists identify six typical phases per procedure, each of which integrates the act(s) with the related documentation; (1) the **initiative** (the start of a procedure which produces records, like

an application or claim); (2) the **inquiry** (the collection of information needed to evaluate the situation and producing records, like a survey and estimate); (3) the **consultation** (the collection of opinion and advice based on the information accumulated and producing records, like minutes and discussion papers); (4) the **deliberation** (decision-making producing records like contracts); (5) the **deliberation control** (independent review of the record, its form, substance, and the deliberation process); and (6) the **execution** which formalizes the transaction and its binding nature, and includes all of the transmissions, registrations, and other related matters associated with the process.

Documentary form

Form is the set of rules governing the representation by which an act is documented or a message is conveyed. It is the abstraction of what constitutes a type of record from the particulars of any given record. It is a standard template for all records of the same type that can be used to **analyze records to determine** their **nature**, **provenance**, and **trustworthiness**. **Extrinsic** elements of form are the formal characteristics that determine the appearance of the record and its effect. **Intrinsic** elements are the unique parts of the record that associate it with a particular act or transaction and make it complete.

Extrinsic elements of form

These consist of the **medium** (i.e., the physical carrier of the record which is studied in terms of the material, manner of preparation, watermarks, shape, size, edging, rulings, etc.), **script**,(i.e., fonts, layout, paragraphing, punctuation, abbreviations, and initialisms), **language** (i.e., style, formulas, and tenor of discourse), **special signs** (i.e., symbols identifying persons involved with the record, like logos, heraldic markings, mottos, stamps, or drawings which are considered key to provenance), **seals** (which are examined for material, size, shape, typography, legend, and affixation method as indicators of origin and authority of the record), and **annotations** (i.e., additions made after the record is complete).

Annotations come in three types; (1) those added **at the conclusion of the production of the record** (e.g., the annotation on a record of a record identifier placed in a separate register, with

relevant page, date, etc.); (2) those added during the record's **use after creation** (e.g., mention of the decision made or further actions to be carried out, dates of hearings or readings, markings like “urgent”, etc.); and (3) those added **in record keeping activities** (e.g., a registry number, classification markings, Dunns numbers, metadata, etc.)

Intrinsic elements of form

Records present a structure with three sections; (1) **protocol**, containing the administrative context of the act (e.g., it's place, time, date, subject, persons participating, etc.); (2) **text**, containing the action or message and its motivation, circumstances, or conditions; and (3) **eschatocol** (containing the means used to validate the record, such as the signature of the author, witnesses, and countersigners). There are standard forms for different sorts of records, such as contracts, court orders, etc. Even for day-to-day records made by digital systems, there are often many such elements. For example, most messaging systems include intrinsic elements of form like header areas, and extrinsic elements of form, like timestamps and digital signatures.

Archival bond

The archival bond is the relationship between records generated in the course of activity. It is **originary** (i.e., it exists from the moment a record is created); **necessary** (i.e, there is no record without it); and **determined** (i.e., defined by the function of the record in the business activity it is part of). A good way to think of this is in context of a whole archive evolving over time through sequences of acts by multiple persons. The archive is a context in which a record is understood. In a paper system, a record may be placed at a location in a filing cabinet based on its characteristics (e.g., last name of originator, sequence of arrival, etc.). The record becomes a record when placed there. If it's not where its supposed to be, it's just a document, deprived of meaning, and not to be relied upon.

In a computer system, a file system may be thought of as an archive into which drafts (e.g., files) are placed when first saved, as modified over time, and when perfected (as records) along with annotations (e.g., metadata). if there is no such storage, there is no such record. Lacking the proper context of the archive, according to

the archival postulate that a record is made up of a document and the whole of its relationships, diplomatics does not admit that a record can be fully understandable.

Summary of diplomatics

Diplomatics has, as its foundations, a great deal of history and practice that has been accepted by the legal system for thousands of years. As such, it is an outstanding model to start to deal with the issues of DFE examination. The terminology is well and widely defined and used in the legal system, and it is informative and useful for defining the science of digital forensics for these reasons.

Electrical and computer engineering

Computers would not exist in the form they are today if it weren't for the efforts underlying electrical and computer engineering. In simple terms, computers are engineered systems that apply the science and discipline of electrical and computer engineering to create highly reliable mechanisms for storing, processing, reading, writing, and communicating digital information.

The way engineers do this is by applying scientific results relating to the theory of electromagnetic energy and properties of materials to create complex composites of components organized so as to perform specific functions under specific conditions. The conditions are part of the specifications used by engineers to design computers. For example, most computer components are designed to operate over temperature ranges of a few tens of degrees centigrade, with input voltages and currents limited to specific ranges of values, at clock rates of some range of millions of cycles per seconds, with limited shaking, radiation, bending, and so forth.

The engineering typically involves using arrangements of substantial numbers of electromagnetic particles (electrons and protons) and sometimes photons, to represent binary digits (bits). These bits and the mechanisms that store and process them are designed to have two stable states, so that every stored value is in one or the other of those states under normal operating conditions, when it is ready to be used for the next step in the machine execution process. The mechanisms use the redundancy of the large numbers of particles along with the bistable nature of the design and added redundancy at the component and composite

level to provide a highly reliable way to store and process these bits. These mechanisms are combined to make finite state machines that will be discussed in more depth later.

The notions of causality and the science and engineering practices involved in these areas are well understood relative to the things that these systems are being designed to do, and for this reason, we have highly predictable automated mechanisms that support a scientific basis for making claims about what a “bit” is, how it is represented in the physical world, and the relationship between the physical world of these engineered devices and the logical world of digital systems in the sense of finite state automata.

The mathematics of computation

At the level the finite state automata that are produced by the engineered nature of computer systems, there is also a significant body of research that supports basic knowledge of the workings and mechanisms of these finite state machines.

These results codify a mathematical framework for extending the hardware into functional mechanisms that implement software in the form of operating systems, libraries, applications, and content. Thinking in terms of the context of diplomatics, the software supported by the hardware has the capacity to implement a full range of mechanisms to provide for controlled procedures resulting in a proper system of records. But the reality of today's systems and software are that they don't do this. Rather, the state of the art in software fails to recognize the role of the historical science of diplomatics in favor of an anarchist approach to the production of content and mechanisms.

In this system, the role of the DFE examiner is very different from that in the far more highly organized arena of making and verifying government documents. Trustworthiness cannot normally be readily identified based on procedure, but rather must be established by examination of reliability, authenticity, accuracy, and authentication of records through the more complex review of documentary form in context.

Because the complexity of computation has been deeply explored, particularly in the areas related to analysis of records for similar

properties, we have the basis for applying these results to the science of DFE examination. But before we do that, we must start by codifying the elements of computational complexity in terms of the science and engineering disciplines of electrical and computer engineering, so that the necessary background and context is in place to evaluate the properties of medium script, language, special signs, seals, and annotation and do so in the context of the archival bond.

The basics of these fields are explored and the art with respect to these issues in terms of computation are tracked as they relate to DFE examination throughout this book in the hopes of bringing clarity to the relationship between these areas.

An epistemology for digital forensics

Epistemology studies the nature of knowledge, its presuppositions, foundations, extent, and validity.[10,11] For DFE examination, some basics may be reasonably assumed for the purposes of creating a science. Here are some of the epistemological issues already identified.

Digital evidence consists entirely of sequences of binary values that we call bits. Thus, in this limited field, we do not deal with the physical nature of normal space, but we operate in a very different space.

The physics of DFE is different than that for matter and energy, and thus the normal assumptions that are made with respect to the way the world works do not apply, or don't apply in the same way, to DFE. Substantial differences include, without limit:

> DFE has observation without alteration and duplication without removal.
>
> Computational complexity limits what can be done with what resources in what time frame - a different "speed of light".

10 http://www.merriam-webster.com/dictionary/ defines it as: "the study or a theory of the nature and grounds of knowledge especially with reference to its limits and validity";

11 http://www.thefreedictionary.com/ - defines it as: "The branch of philosophy that studies the nature of knowledge, its presuppositions and foundations, and its extent and validity.".

Unlike most physical evidence, which is very often transfer evidence and sometimes trace evidence, DFE is always trace evidence, but essentially never transfer evidence.

DFE is normally latent in nature in that it can only be observed through the use of tools. This then implies a multitude of requirements surrounding those tools and their use.

In a "scientific" approach, the theories are not casual theories, but "scientific theories". That means that:

> They are constructs that are testable.
>
> Refutation can destroy a theory, but finite confirmations cannot "prove" it. They can only confirm it.
>
> In normal science, scientific theories change slowly. Once accepted, they only change because of rare and substantial changes in the scientific community's understanding of the underlying nature of the World.

The "theories" of DFE lead to a physics of digital information. They are largely based on widely accepted mathematical knowledge, but some are still conjectures from computer engineering, computer science, finite mathematics, and related areas.

A quick introduction to information physics

The physics of digital information is significantly different than the physics of the physical world we deal with on a day-to-day basis. There are many differences between these worlds and many of them are described in more detail in Chapter 3. The basic reason for this is the science and engineering disciplines involved in electrical and computer engineering. Because these engineering disciplines are so effective at constraining the mechanisms to deal with only binary values, they display emergent characteristics that are quite different from normal physical systems, when viewed at the level of the digital artifice they create.

To get a sense of the sorts of differences we face, many of the underlying assumptions of the physical world, such as smoothness, continuous space, the notion of transfer, continuous time, and even the speed of light, are all very different in the digital world, and in

many cases, simply don't hold true. The implications of these differences are, in some sense, profound.

Input sequences to digital systems produce outputs and state changes as a function of the previous state. To the extent that the state or outputs produce stored and/or captured bit sequences, these form traces of the event sequences that caused them. Thus the definition of a trace may be stated as: "A set of bit sequences produced from the execution of a finite state machine." (FSM)

We think of the physical space we live in as a space that diverges with time. An initial condition in history produces a set of possible future outcomes. When looking at a physical trace, at least theoretically, we could identify a unique historical event sequence that produced such a trace. But the digital space converges with time, so that instead of the one to many relation that we see in the physical world, we see a many to one relation in the digital world. That means that a very large number of potentially very different input sequences and initial states may produce identical traces. Almost any digital trace we identify could be the result of a large number of different historical event sequences, and the number of those sequences increases with the passage of time (i.e., the execution of FSMs). Thus traces from digital mechanisms are not, in general, unique as to the input sequences that produced them.

Another less mathematical sort of problem is the relationship between the unlimited granularity of the physical world in both time and space and the finite granularity of the digital world in both time and space. Because of this difference, at the interface between the physical and digital world there is a discontinuity, near which small differences are exaggerated, and far from which larger differences are ignored.

The limited sensor and actuator capacity of the devices that convert between the digital and physical world also largely prevent the exchange of a wide variety of information that is potentially probative, and make a wide variety of forgeries at the interface far easier than they might otherwise be. This then also implies that input sequences do not directly demonstrate what non-digital events sequences may have produced them. As a result, additional effort is required to attribute traces to real-world causes, and

forgery is potentially far easier in the digital space than in the physical space.

> **The larger implication of these examples is that digital forensic evidence is the result of processing with FSMs, and that processing inherently limits the potential utility of that evidence for providing probative information regarding real-world events.**

DFE examiners must take these limitations into account when undertaking their examinations, and when testifying about the results of those examinations. If they fail to do so, they will tend to produce results that are inconsistent with the facts and fail to meet the rigors of a scientific approach, and their testimony may lead to poor decisions by the court.

These limitations are due directly to the limits of DFE and the methodologies used to understand and work with it. For example, the manner in which examiners are limited in their ability to examine seals is reflected in the fact that digital seals are all binary values, which can easily be reproduced, unlike physical seals.

A quick introduction to the standard model

The model of DFE examination is related to an overarching model of digital forensics that is detailed in Chapter 4. It can be briefly codified in mathematical terms as follows:

Laws L:{l1, ..., ln}, R:{r1, ..., rm}, LxR→[F|T],

Violations LxR→V

Claims E: $\{E_1, ..., E_o\}$

Hypotheses H: ∃h∈H, h∈E

Events E: [∃e, e∈E*] that demonstrate claims [$\forall E_x \in E$, E_x: ($e_{x1} \in E^*$, ..., $e_{xp} \in E^*$)]

Traces T:(t_1, ...,t_q),

Internal consistency C:TxT→[-1...1]

Demonstration consistency D:TxE*→[-1..1]

Forensic Procedures P:{p1, ..., pn}, $\forall p \in P$, $p \rightarrow c \subset C$, $p \rightarrow d \subset D$, $p \rightarrow c \not\subset C$, $p \rightarrow d \not\subset D$

Resources R:(T,$,C,E)

Schedule sequence S:(s1, s2, ...),$\forall s \in S$, s:(l⊂L, r⊂R, h⊂H, e⊂E, t⊂T, c⊂C, d⊂D, p⊂P, r⊂R, t, t')

In essence, the legal claims constitute a set of runs through the elements of laws that produce violations. This can be conceptualized as a partially ordered set (POset). The events and traces are the things that are evaluated to determine the outcome of the legal matter, and they form the basis for the claims that demonstrate the runs through the violation POset. If the events and traces are consistent with an unbroken path through the POset, a violation is indicated, and if not, inadequate indications for a violation are present. If T and E are inconsistent with the POset, then they may act to sever all of the paths forming violations, in which case adequate basis may be present to definitively demonstrate that no such violation is justified. To the extent that T and E are internally or demonstrably inconsistent, C and D may be used to show that the evidence or the claims are less probative, or potentially even prevent their admission into the matter.

The fundamental theorem of DFE examination in this model may be stated in relatively simple terms:

What is inconsistent is not true.

DFE examination then consists largely of testing hypotheses related to the POset that forms V as demonstrated by T and E so as to try to refute them by showing that they produce inconsistencies. This then also implies some things about language and usage.

WARNING: *Appearances may be deceiving. Things that seem inconsistent may not actually be inconsistent.*

Something did happen. The question is what? A theory of the case consistent with the evidence is highly desirable. Consistency and inconsistency are demonstrated by logic and the theories associated with the physics of digital information. So, for example, given that a claim is based on an event e1 causing a trace t1,

events and/or traces showing that t1 happened before e1 would be inconsistent with the claim of causality because information physics demands that cause precede effect.

There are a range of consequences of this model related to things like (1) the sizes of the model components, (2) available computing power and its implication on thoroughness, (3) limitations due to resources and schedule, (4) limits of currently available procedures, (5) legal limitations on what can be used how, and (6) probative versus prejudicial value and its relationship to consistency and related matters. In the example above, the refutation is based on traces and events that may themselves be problematic as well. Thus C and D are defined over a range.

In many cases, because of the limits of DFE examination as described here and elsewhere, more certainty is desired. There are two general classes of approaches that have been identified for higher surety in DFE examination results; (1) identifying additional traces or procedures to gain additional demonstrations of consistency or inconsistency, and (2) identifying redundant paths to prove hypotheses so that even if some paths are less certain or are able to be cut, the overall hypotheses remains intact. These issues are also covered in the model as well as in previous models.

Careful use of defined terms

No matter how many tests are performed, except for special cases, DFE results cannot prove a broad claim true.[12] The best that can be done is to show that tests failed to refute hypotheses and to show the extent to which tests were thorough. Reasonably, the most authoritative claim in [opposition] support of a hypothesis regarding DFE is:

> "The results of [the tests I did] were [in]consistent with [the hypotheses]."

To the extent that some set of these statements then combine together with logical reasoning, an overarching statement may be made with regard to the claims, perhaps of the form:

12 K. Popper, The Logic of Scientific Discovery (1959), Hutchins and Company, London. ISBN10: 0415278449.

> Based on [the basis], I found [traces and events] to be [in]consistent with [claim(s)].

Or in some cases, when this is true:

> In my examinations of [traces and events], everything I found was consistent with [claims] and nothing I found was inconsistent with [claims].

On the other hand, a single refutation disproves a hypothesis, and the least that can be reasonably said if such a refutation is identified is something like:

> "The [procedures I performed] demonstrate that [traces and events] are [inconsistent with / refute] [the hypothesis]."

Thus the methodology of the science of DFE when working on ay particular matter consists of:

- Devising testable hypotheses ($h \in E$)
- Testing those hypotheses against the evidence (T and E) using forensic procedures (P) and logic to determine type C and D consistency by attempting to refute the hypotheses.
- Making properly limited statements about the results of those tests, typically using wording such as that identified above.

There are some other wordings that may apply in other circumstances, and some of the more commonly misused ones are identified here, along with definitions suited to use by the DFE examiner.

> **suggests**:= imply as a possibility ("The [traces / events] suggests ...") - calls to mind - propose a hypothesis or possible explanation.
>
> **indicates**:= a summary of a statement or statements or other content codified ("His statement indicates that ...") OR a defined set of "indicators" are present and have, through some predefined methodology been identified as such ("The presence of [...] (smoke) indicates [...] (fire)")
>
> **demonstrate**:= exemplify - show - establish the validity of - provide evidence for ("The reconstruction demonstrates that ...")

correlates:= a statistical relation between two or more variables such that systematic changes in the value of one variable are accompanied by systematic changes in the other as shown by statistical studies ("Based on [statistical analysis method(s)], the use of the "KKJ" account is correlated (p=95%) with ...")

match:= an exact duplicate ("These two documents have matching publication dates, page counts, ...")

similar::= a correspondence or resemblance as defined by specified and measured quantities or qualities ("The 18 files were similar in that they all had syntax consistent with HTML, sizes under 1000 bytes, ...")

relate:= a defined and specified link ("The file system is related to FAT32 in that FAT32 was derived from ...")

associate:= make a logical or causal connection with basis provided ("I associate these bit sequences with program crashes because ...")

By the careful use of these terms and their consistent application, the field of DFE examination may move forward more quickly, and peer reviews undertaken in the field may be able to create a body of work that is meaningful across time and endeavors. But if, as a field, DFE examination is inconsistent, or if the peer review process fails to force compliance with such terminology, then the science is unlikely to proceed as a normal science or at a rapid pace.

The tools of the trade

As an area of science, DFE examination has a relatively small number of peer reviewed and repeated scientific experiments. In fact, well read readers might be hard pressed to think of more than what they can count on one hand. Today, most experiments:

- are of very limited applicability,
- are not focused on building a fundamental understanding,
- don't meet the standards of scientific rigor expected in other fields, and
- are oriented toward confirmation rather than refutation, which makes them dubious as science.

Furthermore, there is a methodological challenge associated with experiments for several reasons.

DFE is latent, and therefore, experiments require tools. Of course this means that experiments are limited by tools, and like any other area of science, the examiner must understand the limits of the tools in order to understand the limits of the experiments. This, in turn, leads to the need to have a methodology to evaluate tools. Without such a methodology, regardless of that the tools may indicate, the results cannot be properly interpreted.

A reasonable scientific methodology for understanding tools might start with the development of an error model. There are error models for digital systems that have been around for a long time, and they may well be applicable. But examiners need to start applying them as a part of the scientific endeavor, or they will never know how useful their tools are.

Examiners must understand how to calibrate tools, how to test tools, and must create a systematic approach to doing so. Calibration processes typically involve validation with known samples, which is something that can be readily done in most cases, and the testing process typically involves verification of some sort, which in the case of software, normally involves mathematical proofs or tests that verify results against the error models. Again, this is an areas where DFE examination, as a field, has failed to produce. Redundancy through independent result verification may provide an alternative in cases where no well-defined testing methodology and practice is in place.

Regardless of how "good" a tool is, it must be properly used, the results must be meaningfully interpreted, and the limits of the tools must be understood. This implies that the examiner must have knowledge, skills, experience, training, and education suited to the use of the tools they apply. As a field, DFE examination has too few advanced students and teachers and, as a result, produces small numbers of extremely niche "experts" that are of limited utility. There are many niche experts who can potentially speak to very narrow domains. But there are also expert claimants who claim expertise beyond their actual knowledge, skills, education, training, and experience. To few real experts in DFE as a field exist today.

Tools and their limitations are discussed throughout Chapters 5-9, but are particularly emphasized in Chapter 9.

Presentation

Another major issue with tools today is the manner in which they present results, both in support of the examination process, and when results of examination are presented in reports or in front of judges and juries. Chapters 5-9 discuss presentation, but presentation is not represented as a subject in and of itself in this book.

Presentation is intimately tied to, but not directly part of, examination. Because DFE is latent, presentation is always necessarily an issue. For the examiner in examining results of experiments, the results must be presented to the examiner by the tools. For the jury in understanding the evidence and examination results, presentation is again fundamental. For the judge in evaluating admissibility, the same is true. For the opposition in evaluating expert reports, the presentation is just as critical. Today, however, there is no standard for even presenting the most common representations of DFE. Even something as simple as presenting a text file is fraught with potential errors.

Different ways of presenting the same information may lead to different interpretation and outcomes. As a simple example:

> Plaintiff's sworn statements are inconsistent with the evidence.
>
> If Plaintiff's sworn statements are to be believed, the evidence is not.
>
> If the evidence is to be believed, Plaintiff's sworn statements are not.

The first of these statements encompasses the second two. The second one seems to say that we can assume the plaintiff is telling the truth but the evidence is false. And the third one seems to say that the plaintiff is lying.

Technical presentation errors are also problematic. For example, the number `l` and the letter `l` are almost indistinguishable, as are the digit O in the letter 0. The spaces at the ends of lines, and the

difference between a leading tab, a leading space followed by a tab, and leading spaces, cannot be seen in normal outputs. In fact, if you read the end of the last paragraph and the beginning of this paragraph and didn't identify both the presence of the trailing space and the substitution of letters for numbers and numbers for letters, you've just demonstrated to yourself the nature of these errors.

When examining the output from widely used and trusted tools, the presentation produced by those tools often fails to aid the examiner effectively in seeing these sorts of differences. In case after case, and in tool after tool, these differences that might allow the examiner to detect inconsistencies, go unseen. As a result, these inconsistencies are commonly missed. Even something as simple as a forensic font™ would largely alleviate these problems, and yet this notion was only first published in 2010.[13]

Clearly, the area of presentation is fundamental, both to the advancement of science in this area, and to the effective use of tools upon which essentially all of the science of digital forensics and use in and evaluation by courts depend.

The state of the science and coverage of this book

When a legal action involving the formalisms of a court system are involved, and that action involves evidence consisting of 1s and 0s (the binary digits), there are specific concerns that have to be addressed in order to provide accurate facts to those who have to make judgements based on facts.

Those concerns include many things that are not discussed in depth in this book, like identification, collection, preservation, transportation, storage, presentation, and destruction of evidence. Except in the beginning of this book and where the discussion necessarily touches on those other aspects of digital forensic evidence (DFE), they will be largely ignored so that this book can focus on the issues loosely called "examination". This book is intended to cover the state of the science in DFE examination, and to the extent feasible, we have been inclusive within the bounds of the book's depth and breadth.

13 F. Cohen, "Fonts for Forensics", IEEE Oakland Conference, SADFE Workshop, May 19-20, 2010.

Assumptions and a perspective

In this book, some general assumptions are made with regard to the examination process for the purpose of simplification and so the book can focus on the issues of examination. In particular, chain of custody issues, issues associated with the physical realization of the digital forensic evidence, and legal issues not specifically associated with the examination process are ignored and assumed to be taken care of somewhere else.

It might be best to think of the examination processes discussed in this book as representative of the manner in which a certified independent laboratory might work, where that laboratory never receives or handles any original evidence, where it never has to appear in court, but where its results must be suitable for presentation in court. The available DFE arrives in the form of CD-ROMs or file transfers, as what we might call a "bag of bits", and through some method that we are not concerned with, it is authenticated as to being the same bag of bits that was sent to the laboratory. It comes along with a set of statements. The legal team that sent the bag of bits wants to know the truth regarding the statements and the bag of bits in terms of whether and to what extent they are consistent or inconsistent with each other, and wants to know how we got these answers and how reliable those answers are.

When making statements in a legal setting, unlike in academic presentations, the term "I" is used to indicate things stated by people about themselves or what they did. In this book, I may make some such statements, while in other cases, statements will be put forth, typically with citations, indicative of what others have done, general conclusions, things we may share, and so forth. It is my intent that these differences be understood by the reader in context.

What this book covers in depth

In this book, the general area of examination is broken down into analysis, interpretation, attribution, and reconstruction.

- Analysis consists of a set of processes used by the DFE examiner to seek to understand and characterize the evidence relative to the issues in the legal matter. These analytical processes should be well understood and

systematic methods applied in accordance with defined methodologies so that they can be asserted with some degree of reliability to produce the results they are supposed to produce.

- Interpretation consists of taking the results of analysis and producing meaningful statements about what it implies in terms of the technical and legal situation. The interpretation should be done in such a manner that other examiners with proper understanding would reasonably agree to it.
- Attribution consists of drawing conclusions about causes and effects and the links between them. In essence, it asserts that a particular effect is consistent with the effects of a particular cause and that other causes are inconsistent with the effect.
- Reconstruction is a process by which a set of systems or mechanisms similar to the ones known or postulated to be present in the relevant situation are reconstructed and experiments are performed using the reconstruction to show that evidence is or is not consistent with the reconstruction and the assumptions under which it was made.

The general field of digital forensic evidence examination has not been well characterized historically, and in large part, the purpose of this book is to give a framework and specific examples of how examination may be performed so as to meet the rigors of legal systems.

Overview of the book and its overview of the science

This book starts in-depth coverage in Chapter 2 with an overview of digital forensics derived from an article written only a few months before the book was started. This overview provides a basic understanding of the overall challenge of DFE in the larger picture of the legal context, and is intended to set a baseline for understanding the remainder of the book. In many cases, the rest of the book refers back to this chapter and uses concepts associated with it to discuss matters associated with examination.

Chapter 3 is unique to this book as of this writing in that it is, as far as I am aware, the first time that the subject of information physics

in the sense it is used herein has either been identified as a subject of study or defined in any meaningful way. Information physics is a field that has, in my opinion, long been needed, and that has applications well beyond the DFE examination arena. Without it, the DFE examination field will not, in my opinion, make real progress. At the same time, the results in this area are hardly new. For the most part, they are results that have long been known from other fields, but that haven't been gathered together in one place and focussed on one particular application. That is the intent of this chapter, and the results of this chapter are applied throughout the remainder of the book and throughout many related fields.

Chapter 4 produces a theoretical framework for DFE examination by extending previous frameworks and considering their utility in a practical sense for defining a scientific field of enquiry. It examines recent results in attempting to create similar frameworks, and expands and adopts things from those frameworks where they are helpful, abandoning other parts where they are not. The reader should be keenly aware of and read these other works in order to understand their value and limitations and where this work stands in relation to them. The present model, as defined in this chapter, along with information physics, form the basis of the theory of DFE upon which the science of today rests. As such, it is the thing that is to be tested and confirmed or refuted, if the science is to move forward.

Chapters 5, 6, 7, and 8 focus in on the different aspects of the science of examination, as opposed to the overall field of digital forensics. In the view of this book, examination consists of:

- **Analysis**, is largely a mathematical and technique-oriented area that is intended to identify consistencies and inconsistencies based on established methods that are measurable against defined criteria.
- **Interpretation**, is, at its essence, a human endeavor that reconciles the meaning of events and traces with the theoretical underpinnings of the field and the subtleties of legal matters to allow the examiner to make statements about analytical and other results that are meaningful to the case at hand.

- **Attribution** may be characterized as a subset of interpretation associated with determining causality, and is largely about the interpretation of things that lie outside of the digital realm in terms of traces that exist within the digital realm, and as such, it is worthy of study on its own.
- **Reconstruction** is the experimental branch of DFE examination. It fulfills the scientific purpose of allowing for the confirmation and refutation of hypotheses about the matter at hand by experimental demonstration.

Chapter 9 looks more deeply into the tools and processes used in the science of DFE examination, how those tools and processes are scientifically applied, and their use and understanding by the examiner in the legal context.

Chapter 10 closes the book with the state of consensus in digital forensics today and questions about the future.

Moving toward normal science

A quick summary of what we can say today based on the science of DFE examination is, to a close approximation:

> I did X and observed Y
>
> I [did not find / found] X in Y
>
> I found that X is [in]consistent with the claim Y because...
>
> I found that X [suggests/indicates / demonstrates/ correlates with/ matches / is similar to / relates to / associates with] Y because...

Each of these can, if properly undertaken, have a sound basis in the scientific underpinnings described. But the current set of methodologies, processes, and procedures are limited in their validity, testability, reliability, calibration, and basis. There is a lack of strong agreement within the scientific community as to many aspects of the science as presented here.

While most of the results are peer reviewed and accepted within individual communities, the overall collection of results, as a body of science, is not recognized as such. In particular,

- The unifying methodology expressed in regard to the application of information physics to determine consistency is only slowly gaining universal acceptance.
- The model is only one of several currently in use for various limited purposes, and is not widely adopted yet.
- Tools and processes are only explored to a limited extent, with notions of completeness and thoroughness only starting to be defined. Error models have not been adequately applied from existing fields.
- Procedures and their results are limited and the conclusions from such procedures are not formalized or standardized.
- The sources and magnitudes of uncertainty are poorly defined, and confidence intervals for many results simply do not exist.
- Scientific acceptance as measured by surveys and literature reviews shows limited consensus levels for basic concepts in the various communities that make up the field today.

In most cases, the honest and knowledgeable examiner is largely limited to the most basic "I did X and observed Y", with the observation being typically limited to "I [did not find / found] X in Y ". While these are powerful statements that are appropriately used in place of other less sound statements, they are a long way from the level of science that DFE examination has the potential to achieve.

DFE examination is not operating as “normal science” today. While there is a scientific basis for many of the things we may do, and much of it may be reasonably well explained by this book and elsewhere, the overall digital forensics community appears to lack consensus surrounding a common technical language. While there appears to be a level of consensus beyond random levels surrounding basic principles, this can only be discerned when the lack of consensus surrounding language is controlled for. We have the foundations of scientific theories, but too little attention is paid to testing the theories and developing the science further. We have common understandings, but the lack of common language and enforcement of such language in publications makes the

expression of those understandings difficult to clearly discern. A call to address this has gone out, and hopefully, times are changing.[14]

Ultimately, the DFE examination community needs to ask itself some of these questions, as a community:

- What can we build community consensus for?
- What well-defined and consistently used terms should we embrace?
- What well-understood epistemology will we use?
- What theory / methodology will we choose?
- What strong experimental basis will we build?
- What agreed-upon physics will we use and how will we formulate it?
- How will we build a community consensus?
- Is the path outlined here something we want to embrace?
- If not, how should we change it?

The view of this book is that there is now at least one description of a reasonably comprehensive scientific foundation underlying DFE examination. Regardless of its many possible problems and limits, it is a place to start building a normal science and advancing that science in the normal manner. As the field matures, normal science is almost inevitable, but the normalization process is only just beginning today. A community consensus is highly desired, and this book supports and anticipates such consensus in the near future.

14 F. Cohen, "Update on the State of the Science of Digital Evidence Examination", 2012-01-15 Submitted to Journal of Digital Forensic Evidence.

Questions

1. What is the difference between the general term "examination" and the more specific term "analysis" discussed in this chapter?
2. Are there different skills and characteristics likely to be used in examining of digital forensic evidence than in the other aspects discussed here? What are they?
3. This book seeks to look at DFE in terms of a scientific approach. What are the advantages and disadvantages of this approach from the perspective of the reader who has not yet read the book in depth?
4. This book ignores many of the things covered in other books while focussing in on areas that are less well covered elsewhere. To what extent are there obvious interactions between other aspects of the field and the examination area, and how can they be realistically separated? What is the most likely dividing line between them?
5. Given the coverage of this book and the relatively mathematical approach taken, what is your background in the field, and how do you see this book assisting you in moving to another level?
6. What areas do you imagine you might need help in to get through such a book as this, and how will you get the help you need in completing it?
7. In what ways does the use of specific language and providing a basis for your statements help and hurt your examination of DFE?
8. What do you think of picky people in picky processes? Are you prepared to be a picky person in a picky process, and if not, is this the wrong field to be in?
9. Answer the questions on the previous page. How will you help to move the field forward based on your answers?
10. When do you think DFE examination will reach the level of normal science?

2 An overview of digital forensics

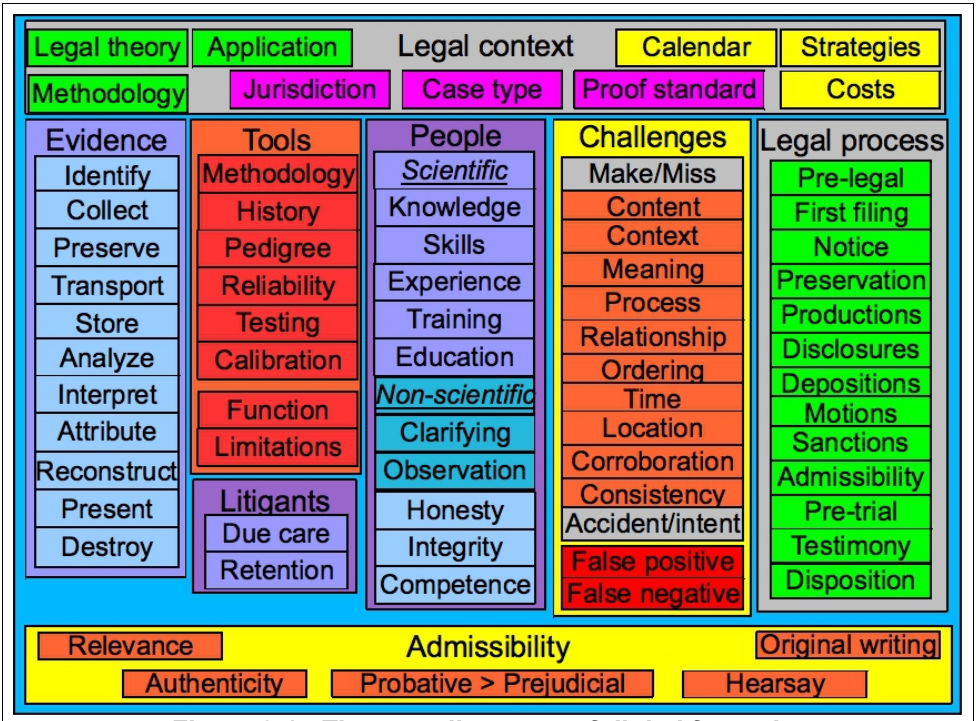

Figure 2.1 - The overall context of digital forensics

Figure 2.1 shows the overall view of digital forensic evidence that this book takes.

Introduction

Digital forensic evidence consists of exhibits, each consisting of a sequence of bits, presented by witnesses in a legal matter, to help jurors establish the facts of the case and support or refute legal theories of the case. The exhibits should be introduced and presented and/or challenged by properly qualified people using a properly applied methodology that addresses the legal theories at issue. The tie between technical issues associated with the digital forensic evidence and the legal theories is the job of expert witnesses.

Exhibits are introduced as evidence by one side or another. In this introductory process, testimony is presented to establish the

process used to identify, collect, preserve, transport, store, analyze, interpret, attribute, and/or reconstruct the information contained in the exhibits and to establish, to the standard of proof required by the matter at hand, that the evidence reflects a sequence of events that is asserted to have produced it. Evidence, to be admitted, must be shown by the party attempting to admit it, to be relevant, authentic, not the result of hearsay, original writing or the legal equivalent thereof, and more probative than prejudicial. Assuming that adequate facts can be established for the introduction of an exhibit, people involved in the chain of custody and processes used to create, handle, and introduce the evidence testify about how it came to be, how it came to court, and about the event sequences that may have produced it.

Digital forensic evidence is usually latent, in that it can only be seen by the trier of fact at the desired level of detail through the use of tools. In order for tools to be properly applied to a legal standard, it is normally required that the people who use these tools properly apply their scientific knowledge, skill, experience, training, and/or education to use a methodology that is reliable to within defined standards, to show the history, pedigree, and reliability of the tools, proper testing and calibration of those tools, and their application to functions they are reliable at performing within the limitations of their reliable application. Non-experts can introduce and make statements about evidence to the extent that they can clarify non-scientific issues by stating what they observed.

Digital forensic evidence is challenged by identifying that, by intent or accident, content, context, meaning, process, relationships, ordering, timing, location, corroboration, and/or consistency are made or missed by the other side, and that this produced false positives or false negatives in the results presented by the other side.

The trier of fact then must make determinations about how the evidence is applied to the matter at hand so as to weigh it against and in conjunction with all of the other evidence and to render judgements about the legal matters that the evidence applies to.

The legal context

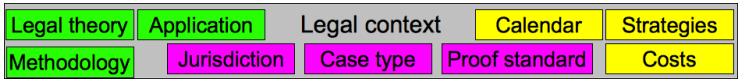

Figure 2.2 - The legal context

Digital forensic evidence is and must be considered in light of the legal context of the matter at hand. This context includes, without limit:

- The legal matter determines the jurisdictions involved and thus the applicable laws and legal processes, the legal theories, methodologies, and applications of those methodologies that will be accepted, the requirements for admissibility of evidence, the requirements for acceptance of expert witnesses, the standards of proof, and many other similar things that impact the digital forensic evidence and its use.
- The nature of the case, whether it is civil or criminal, and sub-distinctions within these broad categories, affects the standards of proof and admissibly, the rules of evidence, the rules for trials, and many other aspects of what can and cannot be used in the legal matter and supported or refuted through digital forensic evidence.
- Limitations on elements of the case such as searches and seizures, which may be real-time or after the fact, compulsory or permission, and limited in various ways so as to prevent them from becoming "fishing expeditions" are informed by and help to form the context within which the digital forensic examiner must operate.
- Procedural requirements of legal cases may constrain certain arguments and evidence so that it can only be used at particular times or in particular types of hearings.
- The calendar is often daunting in legal matters, and in many cases there is very little time to do the things that have to be done with regard to digital forensic evidence. The calendar of the case may also impact the sequence in which evidence is

dealt with, and this may result in additional complexities relating to the ordering of activities undertaken.

- Cost is an important factor because only finite financial resources are available. While there may be an enormous range of examination that could be undertaken, much of it may not be undertaken because of cost constraints.
- Strategies and tactics of the case may limit the approaches that may be taken to the digital forensic evidence. For example, even though some sorts of examination may be feasible, they may be potentially harmful to the side of the case the forensic examiner is involved in, and therefore not undertaken by that side.
- Availability of witnesses and evidence is often limited. In some cases evidence may only be examined in a specific location and under specific supervision, while in most cases, witnesses are only available to the attorneys during limited time frames and under limited circumstances. For the opposition to the party bringing the witness, these may be very limited and restricted to testimony under oath in depositions and elsewhere.
- Stipulations often limit the utility and applicability of digital forensic evidence. For example, if there is a stipulation as to a factual matter, even if the digital forensic evidence would seem to refute that stipulation, it can be given no weight because the stipulation is, legally speaking, a fact that is agreed to by all parties and therefore cannot be refuted.
- Prior statements of witnesses often create situations in which digital forensic evidence is applied to confirm or refute those statements. In these cases, the goal is to find evidence that would tend to refute the statements and thereby make the witness and their prior testimony incredible.
- Notes and other related materials are potentially subject to subpoena in legal matters, and therefore, conjectures on notes, FAXes, and drafts of expert reports as well as other similar material might be discoverable and used to refute the

work of the experts. This tends to limit the manner in which the expert can work without endangering the case for their client.

There are many other similar legal contextual issues that drive the digital forensics process and the work of those who undertake those processes. And without this context, it is very difficult if not impossible to do the job properly. While it is the task of the lawyers to limit the efforts of the digital forensics evidence workers in these regards, it is the task of the workers to know what they are doing and how to do it properly within the legal context.

Those who engage in work related to digital forensic evidence must understand these issues at a rudimentary level in order to be useful to the legal process, and they must understand these issues and be willing to work within the context of the legal system and the specifics of the matter at hand in order to work in this area.

The processes involved with digital forensic evidence

While there are many other characterizations of the processes involved in dealing with digital forensic evidence (DFE), the perspective taken here will assume, without limit, that DFE must be identified, collected, preserved, transported, stored, analyzed, interpreted, attributed, perhaps reconstructed, presented, and, depending on court orders, destroyed.[15] All of these must be done in a manner that meets the legal standards of the jurisdiction and the case.

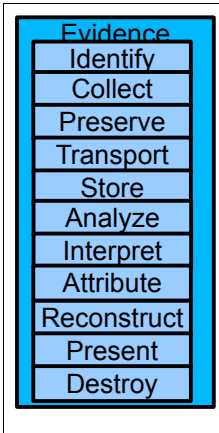

Figure 2.3 - DFE processes in legal contexts

In this book, the focus of attention is on the part of the process involving analysis, interpretation, attribution, and reconstruction, which collectively are called "examination". The examiner examines the DFE through processes and using tools that meet the requirements of the case and ultimately provides results in some meaningful form that are used by the parties in the case for their purposes.

15 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

Identification

In order to be processed and applied, evidence must first, somehow, be identified as evidence. It is common for there to be an enormous amount of potential evidence available for a legal matter, and for the vast majority of the potential evidence to never be identified. To get a sense of this, consider that every sequence of events within a single computer might cause interactions with files and the file systems in which they reside, other processes and the programs they are executing and the files they produce and manage, and log files and audit trails of various sorts. In a networked environment, this extends to all networked devices, potentially all over the world. Evidence of an activity that caused digital forensic evidence to come into being might be contained in a time stamp associated with a different program in a different computer on the other side of the world that was offset from its usual pattern of behavior by a few microseconds. If the evidence cannot be identified as relevant evidence, it may never be collected or processed at all, and it may not even continue to exist in digital form by the time it is discovered to have relevance.

Collection

In order to be considered for use in court, identified evidence must be collected in such a manner as to preserve its integrity throughout the process, including the preservation of information related to the chain of custody under which it was collected and preserved. Recent case law has established that there is a duty to preserve digital forensic evidence once the holder of that evidence is or reasonably should be aware that it has potential value in a legal matter. This duty is typically fulfilled by collecting and preserving a copy of the original evidence so that the actual original media need not be preserved, but rather, can continue to be used. Collection may involve many different technologies and techniques depending on the circumstance.

What is collected is driven by what is identified; however, a common practice in the digital forensics community has been to take forensically sound images of all bits contained within each media containing identified content. This provides the means to then identify further evidence contained within that media for subsequent examination, assuming that the copy of the media was

properly preserved along the way. The problem with this process today is that the volume of storage required has become very large in many cases, and this process tends to be highly disruptive of operating businesses that use these computers in a non-stop fashion. Consider the business impact on an Internet Service Provider (ISP) if they have to cease operations of a computer that would otherwise be in use in order to preserve evidence.

Preservation of relevant log files and audit data is particularly important and should always be identified and preserved. This includes all logs associated with the servers used to send, receive, process, and store the evidence. Failure to do this becomes particularly problematic in cases when the purity of the evidence is at issue. For example, if an exhibit contains some corrupt content, the entire exhibit becomes suspect. If original records are not available to rehabilitate relevant portions of the exhibit, all of the evidence contained in the exhibit may be inadmissible. If there is suspicion of spoliation, the additional log files and related traces will likely be necessary in order to show that redundant information exists that is consistent with the actual creation of the content at issue. Even information such as system crashes and reboots may be critical to a case because corrupt file content may be produced by those sorts of events and without the logs to show what happened when, that corruption may not be able to be reconciled with the need for preservation of the purity of the evidence.

Many cases have hinged on log, audit, and other related data, if only to show that the other digital forensic evidence is real. And case after case today is being lost because of inadequate records retention and disposition policies and processes. Almost any case demands that evidence be properly identified and preserved, and that includes meta-data and log data, both locally and from independent third party sources who have no interest in the matter.

Transportation

Evidence must sometimes be transported from place to place. For example, when collected from a crime scene, the evidence must carefully be moved to a secure location or it may not be properly preserved through to a trial. Digital forensic evidence can generally be transported by making exact duplicates, at the level of bits, of

the original content. This includes, without limit, the movement of the content over networks, assuming adequate precautions are taken to assure its purity during that transportation. Evidence is often copied and sent electronically, on compact disks, or in other media, from place to place. Original copies are normally kept in a secure location in order to act as the original evidence that is introduced into the legal proceedings. If there is any question about the bits contained in the evidence, it can be settled by returning to the original. Facsimile evidence, printouts, and other similar depictions of digital forensic evidence may also be transported, but they are not a good substitute for the original digital forensic evidence in most cases, among other reasons, because they make it far harder, if not impossible, to properly analyze what the original bits were. For example, many different bit sequences may produce the output depictions, and identical bit sequences may produce different output depictions. Care must be taken in transportation to prevent spoliation as well. For example, in a hot car, some forms of digital media, particularly magnetic media, tends to lose bits.

Increasingly evidence is transported electronically from place to place, and even the simplest errors can cause the data arriving to be incorrect or improperly authenticated for legal purposes. Care must be taken to preserve chain of custody and assure that a witness can testify accurately about what took place, using and retaining contemporary notes, and taking proper precautions to assure that evidence is not spoliated and is properly treated along the way.[16]

Storage

In storage, digital media must be properly maintained for the period of time required for the purposes of trial. Depending on the particular media, this may involve any number of requirements ranging from temperature and humidity controls to the need to supply additional power, or to reread media. Storage must be adequately secure to assure proper chain of custody, and typically, for evidence areas containing large volumes of evidence, paperwork associated with all actions related to the evidence must be kept to assure that evidence doesn't go anywhere without being

16 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

properly traced. Many different sorts of things can go wrong in storage, including, without limit, decay over time, environmental changes resulting in the presence or absence of a necessary condition for preservation, direct environmental assault on the media, fires, floods, and other external events reaching the evidence, loss of power to batteries and other media-preserving mechanisms, and decay over time from other natural and artificial sources.

Examination and traces

The theory of trace evidence is, in its most basic form, based on the notion that when two objects contact each other, each leaves something of itself with the other.[17,18] In the context of digital evidence, they theory is somewhat different in that it asserts, at the most basic level, that when events occur in digital systems, bits are affected. The challenge of dealing with the trace evidence is to find the relevant traces, analyze, interpret, and attribute them properly, and be able to characterize what events caused those traces to occur. Examination consists of processes that examiners use to analyze, interpret, attribute, and reconstruct these traces. Analysis, interpretation, and attribution of evidence are the most difficult aspects encountered by most forensics examiners, while reconstruction is used in more complex cases to deal with issues where higher degrees of certainty with regard to more detailed aspects of traces are required.

Analysis

In the digital forensics arena, there are usually only a finite number of possible event sequences that could have produced the evidence; however, the actual number of possible sequences may be almost unfathomably large. In essence, almost any execution of an instruction by the computing environment containing or generating the evidence may have an impact on the evidence.

17 K. Inman and N. Rudin, "Principles and practices of criminalistics: the profession of forensic science", ISBN# 0-8493-9127-4, CRC Press, 2001

18 E. Locard, "The Analysis of Dust Traces", Revue International de Criminalistique I. #s 4-5, 1929, pp 176-249, (translated into English and reprinted in 3 parts in A, J. Police Science, 1930 in V1#3, May-Jun 1930, pp276-298, V1#4 Jul-Aug 1930, pp 401-418, and V1#5 Sep-Oct 1930, pp 496-514.)

Since it is infeasible to reconstruct every possible sequence to find all of the sequences that may have produced the actual evidence in a any particular case, analysts focus in on large sets of sequences of events and tend to characterize things in those terms. For example, if the evidence includes a log file that appears to be associated with a file transfer, the name of the file transfer program included in the log file will typically be associated with common behavior of that program and used as a basis for the analysis. The user identity indicated in the log file may be associated with a human or group, and this creates an initial attribution that can then be used as a basis for further efforts to attribute to the standard of proof required.

Interpretation

Of course the presence of this trace in an audit trail doesn't mean that the program was ever run at all or that the thing the trace indicates ever took place or that the user identified caused the events of interest. There are many possible sequences of events that could result in the presence of such a trace. For example, and without limiting the totality of possible event sequences, the trace could have been placed there maliciously, it could be a trace produced by another program that looks similar to the program being considered, it could have been a trace produced by the program even though the file transfer failed, the trace could have been produced by a Trojan horse acting for the user, or the trace could be there because of a failure in a disk write that produced a cross-link between disk blocks associated with different sorts of traces.

The analyst seeking to interpret the evidence should seek to take into account the alternative explanations for evidence in trying to understand what actually took place and how certain they are of the assertions they make. It is fairly common for supposed experts to make leaps and draw conclusions that are not justified. For example, an analyst might write a report stating something like "X did Y producing Z" where X is an individual or program and Y is an action that produced some element of the evidence Z. But this is excessive in almost all cases. A more appropriate conclusion might be "Based on the evidence available to me at this time, it appears that X did Y producing Z". And of course it helps if some or many of

the alternative explanations have been explored and shown to be inconsistent with the evidence. That's one of the reasons that seemingly irrelevant evidence might be very useful in a legal matter. For example, evidence from system logs might indicate that there were no detected disk errors, system crashes or reboots, or other anomalies reflected in the log files for the period in question, and that therefore, the explanations associated with these sorts of anomalies are inconsistent with the evidence. But without those log files or some other evidence, this conclusion cannot be reasonably drawn.

In networked environments, there are potentially far more sequences of bits that may be relevant to the issues in the matter at hand. As a result, there is potentially far more evidence available, and the analysis and interpretation of that larger body of evidence leads to many more potential analytical and interpretive processes and products. It could be argued that this increases the complexity of analysis exponentially, but in reality, the additional evidence tends to further restrict the number of histories that are feasible in order to retain consistency of interoperation across the evidence. As an example, the file transfer trace identified above might be greatly bolstered or flatly refuted by corresponding traces on remote systems from which the file was asserted to be downloaded and through which the transfer may have come.

Attribution

Analysis, interpretation, and attribution of digital forensic evidence are also reconcilable with non-digital evidence and externally stipulated or demonstrated facts. As an example, if the digital forensic evidence appears to show that person X was present at the local console of a computer in Los Angeles, California two hours after they passed through customs and immigration in London, England, even though the network logs from distant systems show that the transfer took place, it is not a reasonable interpretation to assert that the individual was in Los Angeles. Clearly there is another explanation, whether it is two individuals, a remote control mechanism, alteration of multiple logs in multiple systems, alteration of customs and immigration logs, altered time clocks, or any of a long list of other possibilities. While in some venues, the "don't confuse me with the facts" approach may apply,

in a legal setting, digital forensic evidence should reconcile with external reality.

Anchor events that the analyst can testify to are a good example of the interaction between digital forensic evidence and physical reality. An example of an anchor event is knowledge of time keeping mechanisms on systems that interact with evidence available in the matter at hand. For example, if the analyst operates a system that retains sound records and was synchronized to network time protocol during the period of time at issue, and that system has a record of an email passing through a relevant system that includes time and date stamps, then the time skew between the analysts system and the relevant system provides an anchor in facts that the analyst can use to make more definitive statements about what took place and when. Interpretation of the evidence can then more definitively assert that, based on the personal knowledge of the witness and the records they have of facts relevant to the matter, a particular record is consistent with a time skew of 18 hours. This may even allow the analyst to explain how the individual could have appeared to have been in London at the same time they appeared to have been in Los Angeles.

Reconstruction

In many cases, the relevance of the evidence is specific to hardware and/or software. While many analysts make the assumption that mechanisms operate according to their specifications, in the information technology arena, where digital forensic evidence originates, there are in fact few standards and they are liberally violated all of the time. Documentation is often at odds with reality, versions of systems and software change at a high rate, and records of what was in place at any given time are often scarce to non-existent. Legal cases also often come to trial many years after the actual events that led them to take place, and evidence that might have been present at the time of the incident at issue may no longer be available by the time it is known to be of import.

In these cases, reconstruction of the mechanisms that produced the important traces may be the only available approach to resolving, to a reasonable level of certainty, what actually could and

could not have taken place. For example, if the content of the metadata within a document containing evidence of intent indicates that a particular user identity modified the document on a particular date and at a particular time and that the document was edited for 7 minutes and 23 seconds, but does not show specific modifications made by that individual, and a previous version of the document from an hour earlier written with another user identity does not have the content with the evidence of intent and has an edit time of 5 minutes, and no other documentation exists, then it might appear to be strong evidence that the individual who last wrote the document added the content indicative of intent and did so by editing the document for 2 minutes and 23 seconds.

But this conclusion depends on a set of assumptions surrounding the software in use for editing this document. Even if a current version of this software reliably applies this sorts of metadata, it may be that the version of software in use at the time in question and in the computing environments in question did something quite different. If this is the only evidence of the issue at hand, and the matter is important enough to justify the effort, then a reconstruction of the process by which the digital forensic evidence was created may be necessary to show that the specific version of the software operating in the specific environment at issue could or could not have produced the results contained in the evidence and that other possibilities do or do not exist.

Given that a reconstruction is to be considered, additional determinations must be made. For example, based on the available information, how can a definitive determination be made about the version of the hardware, software, and operating environment be made, and how important is it to precisely reconstruct the original situation down to what level of accuracy and in what aspects? The answer to these and other related questions are tied intimately to the details at issue in the matter at hand.

Presentation

Evidence, analysis, interpretation, and attribution, must ultimately be presented in the form of expert reports, depositions, and testimony. The presentation of evidence and its analysis, interpretation, and attribution have many challenges, but

presentation is only addressed to a limited extent in the literature.[19] Presentation is more of an art than a science, but there is a substantial amount of scientific literature on methods of presentation and their impact on those who observe those presentations. Aspects ranging from the order of presentation of information to the use of graphics and demonstrations all present significant challenges and are poorly defined.

Destruction

Courts often order evidence and other information associated with a legal matter to be destroyed or returned after its use in the matter ends. This applies to trade secrets, confidential patent and client-related information, copyrighted works, and information that enterprises normally dispose of but must retain for the duration of the legal process. Data retention and disposition has extensive literature involving legal restrictions on and mandates for destruction.[20] There are also significant technical issues associated with destruction of digital data. The processes for destruction in legal matters rarely rise to the level required for national security issues; however, the efforts involved in evidence recovery do, at times, go the extremes.[21,22,23]

19 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

20 "The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production", September 2004 Public Comment Draft.

21 "A Guide to Understanding Data Remanence in Automated Information Systems", NCSC-TG-025 - Library No. 5-236,082 - Version-2, available at http://all.net/books/standards/remnants/index.html

22 Craig Wright, Dave Kleiman, and Shyaam Sundhar R., "Overwriting Hard Drive Data: The Great Wiping Controversy". Information Systems Security, 4th International Conference, ICISS 2008, Hyderabad, India, December 16-20, 2008, Proceedings; Series: Lecture Notes in Computer Science Subseries: Security and Cryptology , Vol. 5352 Sekar, R.; Pujari, Arun K. (Eds.) 2008, XIII, 307 p., Softcover ISBN: 978-3-540-89861-0

23 Peter Gutmann, "Secure Deletion of Data from Magnetic and Solid-State Memory", Department of Computer Science, University of Auckland, first published in the Sixth USENIX Security Symposium Proceedings, San Jose, California, July 22-25, 1996.

Expert witnesses

The US Federal Rules of Evidence (FRE)[24] and the rulings in the Daubert case[25] express the most commonly applied standards with respect to issues of expert witnesses and will be used as a basis for this discussion (FRE Rules 701-706). Digital forensic evidence is normally introduced by expert witnesses except in cases where non-experts can bring clarity to non-scientific issues by stating what they observed or did. For example, a non-expert who works at a company may introduce the data they extracted from a company database and discuss how the database works and how it is normally used from a non-technical standpoint. To the extent that the witness is the custodian of the system or its content, they can testify to matters related to that custodial role as well.

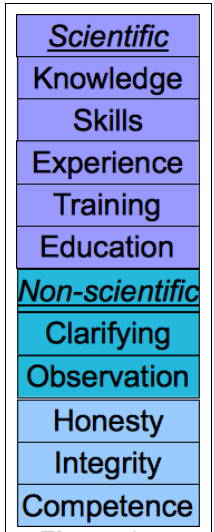

Figure 2.4 - Expertise in legal context

Only expert witnesses can address issues based on scientific, technical, or other specialized knowledge. A witness qualified as an expert by knowledge, skill, experience, training, or education, may testify in the form of an opinion or otherwise, if (1) the testimony is based on sufficient facts or data, (2) the testimony is the product of reliable principles and methods, and (3) the witness has applied the principles and methods reliably to the facts of the case. If facts are reasonably relied upon by experts in forming opinions or inferences, the facts need not be admissible for the opinion or inference to be admitted; however, the expert may in any event be required to disclose the underlying facts or data on cross-examination.[26]

Experts typically have very specialized knowledge about specific things of import to the matter at hand. Anyone put up as an expert

24 The U.S. Federal Rules of Evidence.

25 Daubert v. Merrell Dow Pharmaceuticals, Inc., 509 U.S. 579, 125 L. Ed. 2d 469, 113 S. Ct. 2786 (1993). Theis case dominates in US Federal cases. Daubert extends Frye and also allows accepted methods of analysis that properly reflect the data they rely on.

26 The U.S. Federal Rules of Evidence.

who doesn't have the requisite specialized knowledge is subject to being seriously challenged by competent experts and counsel on the other side. Experts who are shown to be inadequate to the task are sometimes chastised in the formal decisions made by the courts, and such witnesses are often unable to work in the field for a period of many years thereafter because counsel for the opposition will bring this out at trial.

Tools and tool use in digital forensics

Because digital forensic evidence is normally latent in nature, it must be viewed through the use of tools. In addition, tools are used in all phases of evidence examination. In order for tools used in forensic processes to be accepted by the legal system, the tools have to be properly applied by people who know how to use them properly following a methodology that meets the legal requirements associated with the particular jurisdiction.[27] (FRE 701-706)

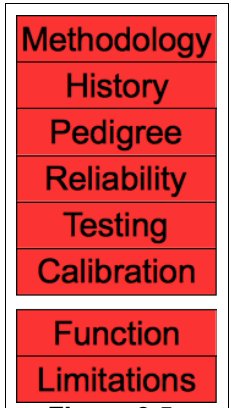

Figure 2.5 - Forensic Tools

One of the key things experts need to know about is the tools that they use. This is because tools are used in almost all tasks associated with DFE examination and tool failures that yield wrong results or tool output that is not properly interpreted leads to opinions and conclusions that may be wrong. One of the main tasks of the DFE expert witness is to identify a meaningful methodology for applying tools to address the legal issues and use that methodology and tools that implement it with known accuracy and precision by examining the evidence and the claims made with regard to the evidence. While some of the claims may be understood with only the experts knowledge, such as assertions that are inconsistent with each other or that fly in the face of current scientific thinking in the field of expertise, most claims in legal matters that involve DFE involve the application of scientific methodologies to evidence through tools.

Tools have history and pedigree that helps to indicate their reliability. Depending on the extent to which the tool provides scientific results that are not obviously verifiable by independent

27 *The U.S. Federal Rules of Evidence.*

means by others, these factors are more important or less important. For example, if a tool, such as the Unix command "wc" counts the number of words, lines, and characters in a file, and the result is used to draw a conclusion about the evidence in the matter, it is something that can be readily confirmed or refuted by any party by simply counting, or in the case of files with many lines, using an independent tool. In this case, the history and pedigree are less important than that the tool has shown reliability at the task it is being relied upon to carry out, that it has been adequately tested, and that it be properly calibrated for its intended use.

Testing of tools is fundamental to their use, and in the field of DFE, an individual brought forth as an expert who has not tested their tools and does not know their function and limitations in adequate detail, is unlikely to be able to withstand cross-examination with regard to those tools or the things those tools are being applied to. This may, ultimately, lead to their disqualification as an expert, or the disregarding of their testimony as not meeting the standards required for credible expert testimony.

While testing of tools may be reasonably done by those who have background in testing of digital systems or by independent bodies, such as NIST, which performs select tests of forensic tools in the United States,[28] calibration must be done by the digital forensics expert prior to and after the use of the tool, assuming that that is required for validation of the tool's accuracy and precision to the level being used for presentation of the results of its use. Very little testing has been formalized in this field for the specific needs of digital forensics, so examiners wishing to be prudent should undertake their own testing programs, and this should be a normal part of the process used in preparing for legal matters where such tools are used. There is a substantial body of well defined knowledge in testing of digital systems, including refereed professional journals, books, conferences, and classes at the undergraduate and graduate level. As an example, the IEEE has had a refereed journal on the subject since 1984.[29]

28 James R. Lyle, Douglas R. White, Richard P. Ayers, "Digital Forensics at the National Institute of Standards and Technology", NISTIR 7490

29 IEEE Design & Test of Computers, issues available starting in 1985 from http://www2.computer.org/portal/web/csdl/magazines/dt#1

The notion of calibration is foreign to many in the digital computer arena, largely because, unlike analog devices which have minor variances due to temperature, pressure, and other physical conditions, digital systems, when working within normal operating ranges, produce either 1s or 0s and do so with very high reliability. Nevertheless, there are calibrations that can and should be done prior to and after the use of DFE tools to validate that what was done did not introduce inaccuracies into the process. As an example, when doing a forensic image of digital media to a different media, the destination media should be pre-configured to a known state so that process failures can be detected. Otherwise, residual data from previous events or from the manufacturing process might be mistakenly intermixed with the new DFE to produce corrupted results. This sort of spoliation has the potential to create enormous problems if the tools and media are not properly calibrated, if error messages are not carefully preserved and taken into account, if contemporaneous logs of the forensic activities are not produced and retained, and if evidence isn't created to verify that the image taken is a true copy of the original evidence. This is similar to the process of cleaning a pipet for a chemical analysis, testing the cleaned pipet to verify that it is free of contaminants, processing the sample, getting the result, then verifying that the pipet is free of contaminants after the sample is analyzed. Failure to undertake such a process would violate standard procedure in chemical testing that has been shown to produce faulty chemical analysis. Similarly, failure to undertake measures to calibrate and verify digital forensic processing of evidence can introduce contaminants or produce faulty digital analysis.

Digital forensic analysis processes often include the creation of special purpose filters, the development of search criteria, and the authoring of small computer programs, sometimes including combinations of scripts written in languages such as the command language of the Unix shell, the Perl language, and other programs written in other languages, and pre-packaged utility programs that come with systems, such as the stream editor "sed", the regular expression string search program "grep", and many other similar sorts of elements. These are commonly combined with tools that retrieve data from Internet sites and process them in various ways to produce outputs that show some analytical result.

When such tools produce results that are readily verified by inspection, such as counts of how many lines of particular types were at particular locations within particular files, the conclusions themselves constitute a testable result that the opposition can challenge and verify. As such, the tools and techniques need not be shown; however, when introducing such evidence, it is incumbent on the producing party to make certain that the results are accurate and precise. To the extent that they are in error and the opposition can demonstrate this, the court will often levy sanctions and potentially exclude the expert and the results from use in court under the admissibility restriction that the results are less probative than prejudicial, the expert witness is not reliably applying a scientific method to the evidence, and that the expert is not in fact adequately knowledgeable or skilled to express scientific opinions to the trier of fact. It is incumbent on experts to provide details of the limits of their results in terms of the limits of accuracy and precision and to not overstate results. For example, when analyzing text files against a format specification, the expert had better understand the extent to which the formal specification is reflected in actual use, and examine results produced for anomalies before declaring the results of the program to be precise and accurate. To the extent that anomalies are detected, they should be explained and the precision and accuracy of results properly characterized.

Challenges and legal requirements

In order to be accepted in a legal proceeding, certain requirements apply to evidence and expert testimony relating to that evidence. On a global level, the most commonly applied standards are similar to the U.S. Federal Rules of Evidence (FRE) and the Daubert decision.[30]

Legal challenges to admissibility under the Federal Rules of Evidence in the US generally go under the following categories. Evidence admitted has to be weighed by the trier of fact in making determinations. Depending on specifics of the circumstances and judicial opinion, evidence may or may not be admitted and its

30 Daubert v. Merrell Dow Pharmaceuticals, Inc., 509 U.S. 579, 125 L. Ed. 2d 469, 113 S. Ct. 2786 (1993).

weight may be expressed by the judge to the jury in formal admonitions for admitted evidence to go to weight.

Relevance: The tendency for evidence to make a fact of consequence determination of the action more or less probable than it would be without the evidence.[31,32,33]

Figure 2.6 - Admissibility

Authenticity: Rules 901-903.[34,35,36] There must be evidence sufficient to support a finding that the matter in question is what its proponent claims. Many illustrative examples are provided, but they are not exhaustive. Examples provided include personal knowledge, non-experts familiar with a unique property such as handwriting,

31 Rule 104 of the Federal Rules of Evidence "(b) Relevancy conditioned on fact. When the relevancy of evidence depends upon the fulfillment of a condition of fact, the court shall admit it upon, or subject to, the introduction of evidence sufficient to support a finding of the fulfillment of the condition."

32 Rule 401 of the FRE: '"Relevant evidence" means evidence having any tendency to make the existence of any fact that is of consequence to the determination of the action more probable or less probable than it would be without the evidence.'

33 Rule 403 of the FRE: "Although relevant, evidence may be excluded if its probative value is substantially outweighed by the danger of unfair prejudice, confusion of the issues, or misleading the jury, or by considerations of undue delay, waste of time, or needless presentation of cumulative evidence."

34 Rule 901 of the FRE "(a) General provision. The requirement of authentication or identification as a condition precedent to admissibility is satisfied by evidence sufficient to support a finding that the matter in question is what its proponent claims."

35 Rule 902 of the FRE. "Self-authentication Extrinsic evidence of authenticity as a condition precedent to admissibility is not required with respect to the following: … (1) Domestic public documents under seal … (2) Domestic public documents not under seal … (3) Foreign public documents … (4) Certified copies of public records … (5) Official publications … (6) Newspapers and periodicals … (7) Trade inscriptions and the like … (8) Acknowledged documents … (9) Commercial paper and related documents … (10) Presumptions under Act of Congress … (11) Certified domestic records of regularly conducted activity … (12) Certified foreign records of regularly conducted activity ..."

36 Rule 903 of the FRE. "The testimony of a subscribing witness is not necessary to authenticate a writing unless required by the laws of the jurisdiction whose laws govern the validity of the writing."

comparisons to known samples by trier or experts, distinctive characteristics, public records, ancient documents, reliable process or system, and methods provided for by statute or rule. Some records may be self-authenticating, such as public documents, certified copies of documents, official publications, and certified records of regularly conducted activity.

Hearsay: Rule 801-802. An out of court statement offered in evidence to prove the truth of the matter asserted is hearsay, but there are many exceptions; most notably business records taken in the normal course of business and relied on for their accuracy and reliability as a matter of course in carrying out that business.

Original writing (best evidence): Rules 1001-1008.[37,38,39,40] To prove content, the original is required unless certain exceptions apply. Exceptions include: (1) originals lost or destroyed, (2) original is not obtainable, (3) the opponent who holds it refuses to produce it upon judicial demand, (4) the content is not closely related to the matter at hand and is thus collateral. Official records are admitted as duplicates. Voluminous records may be represented by statistical samples when they are representative and subject to examination

37 Rule 1001 of the FRE: “(3) Original. An "original" of a writing or recording is the writing or recording itself or any counterpart intended to have the same effect by a person executing or issuing it. An "original" of a photograph includes the negative or any print therefrom. If data are stored in a computer or similar device, any printout or other output readable by sight, shown to reflect the data accurately, is an "original". (4) Duplicate. A "duplicate" is a counterpart produced by the same impression as the original, or from the same matrix, or by means of photography, including enlargements and miniatures, or by mechanical or electronic re-recording, or by chemical reproduction, or by other equivalent techniques which accurately reproduces the original.”

38 Rule 1002 of the FRE: “To prove the content of a writing, recording, or photograph, the original writing, recording, or photograph is required, except as otherwise provided in these rules or by Act of Congress.”

39 Rules 1003 of the FRE: “A duplicate is admissible to the same extent as an original unless (1) a genuine question is raised as to the authenticity of the original or (2) in the circumstances it would be unfair to admit the duplicate in lieu of the original.”

40 Rule 1007 of the FRE: “Contents of writings, recordings, or photographs may be proved by the testimony or deposition of the party against whom offered or by that party's written admission, without accounting for the nonproduction of the original.”

of the originals out of court. When the admission of other evidence depends on facts in this evidence, the court makes the determination, otherwise it goes to weight. When the issue is whether (a) the asserted content ever existed, (b) another piece of content admitted produced it, (c) the evidence in question accurately represents the original, the trier of fact determines it.

More prejudicial than probative: Rule 403 (quoted above). Evidence may be excluded if its probative value is substantially outweighed by the danger of unfair prejudice, confusion of the issues, or misleading the jury, or by the considerations of undue delay, waste of time, or needless presentation of cumulative evidence.

Scientific evidence (expert testimony): Rules 701-706, Frye, Daubert. Non-expert testimony is only admitted if it is (a) rationally based on the perception of the witness, and (b) helpful to a clear understanding of the witness' testimony or the determination of a fact in issue, and (c) not based on scientific, technical, or other specialized knowledge within the scope of expert testimony. (Rule 701) A witness qualified as an expert by knowledge, skill, experience, training, or education, may testify in the form of an opinion or otherwise, if (1) the testimony is based on sufficient facts or data, (2) the testimony is the product of reliable principles and methods, and (3) the witness has applied the principles and methods reliably to the facts of the case.[41] If facts are reasonably relied upon by experts in forming opinions or inferences, the facts need not be admissible for the opinion or inference to be admitted; however, the expert may in any event be required to disclose the underlying facts or data on cross-examination. Recent changes to the Federal Rules of Civil Procedure[42] (FRCP) require that the basis for expert opinion be fully included in expert reports, a change that is vitally important to allowing them to be evaluated scientifically.

41 Rule 702. of the FRE: “Testimony by Experts If scientific, technical, or other specialized knowledge will assist the trier of fact to understand the evidence or to determine a fact in issue, a witness qualified as an expert by knowledge, skill, experience, training, or education, may testify thereto in the form of an opinion or otherwise, if (1) the testimony is based upon sufficient facts or data, (2) the testimony is the product of reliable principles and methods, and (3) the witness has applied the principles and methods reliably to the facts of the case.”

42 See: http://www.utd.uscourts.gov/forms/civil2009.pdf as of Dec 1, 2009.

The Daubert case[43] dominates in US Federal cases. Frye[44] may apply in many states for non-Federal cases. The Frye standard is basically: (1) whether or not the findings presented are generally accepted within the relevant field; and (2) whether they are beyond the general knowledge of the jurors. Daubert also allows accepted methods of processing that properly reflect the data they rely on.

In order to be admitted, digital forensic evidence must survive challenges to relevance, authenticity, its hearsay nature, the original writing requirement, must not be far more prejudicial than it is probative, and must be introduced and analyzed by people who meet standards. It is incumbent on the party introducing evidence to meet these criteria and on the party challenging to oppose based on these criteria and to do so in a timely fashion as part of the legal process. Experts can help make this happen by identifying all lines of challenge and providing analysis, interpretation, advice, knowledge, and skills to help create the conditions for challenges.

It is generally better to make as many such challenges as possible under the theory that if any challenge succeeds it may get evidence disallowed and the more such challenges are presented, the less weight and credibility the evidence will have. Lawyers may not be able to use all of the things that you find as an expert, and time or monetary limits may prevent you from doing as thorough a job as you would like to do, but you can only do what you can do.

It is also important to note that the experts are subject to challenges. If they make too many mistakes, if they are unreliable, if they use techniques that are not in the scientific literature or widely known and used, or if they lack the skill, knowledge, training, experience, or education necessary to qualify them, they can be disqualified along with much of their work.

Last, but by far not least, it is critical to understand that these are not hard and fixed rules that are uniformly applied according to some strict algorithmic formula. Judges and juries are people and

43 Daubert v. Merrell Dow Pharmaceuticals, Inc., 509 U.S. 579, 125 L. Ed. 2d 469, 113 S. Ct. 2786 (1993).

44 Frye v. United States, 293 F 1013 D.C. Cir, 1923 This may apply in many states for non-Federal cases. The Frye standard is basically: (1) whether or not the findings presented are generally accepted within the relevant field; and (2) whether they are beyond the general knowledge of the jurors.

subject to all of the human failings and amazing human capabilities that are inherent in the human species. They have beliefs, points of view, they make cognitive errors, and have likes, dislikes, and biases of all sorts. No matter how hard they try, they may not be able to abandon all of these as triers of fact, and they are not supposed to.

In cases where there is a lot at stake for the parties involved, DFE is likely to be challenged in significant ways. The basic challenges to DFE can be made to a greater or lesser extent at every step of the process, for every item of evidence, and for every witness presented. The challenges may be thought of in terms of a specific set of known fault types that form a fault model.[45]

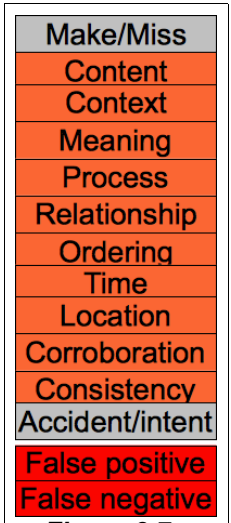

Figure 2.7 - Challenges to DFE

Make or miss faults

In the identified model, faults are characterized as errors of omission, commission, or combinations thereof, sometimes called errors of substitution. Errors of omission are also called "miss" faults because they miss an evidence identification, collection, preservation, transportation, storage, analysis, interpretation, attribution, reconstruction, presentation, or destruction (process) step or miss content, context, meaning, relationship, ordering, time, location, corroboration, or consistency results. Errors of commission are also called "make" faults because they introduce evidence process steps that should not be present or assert content, context, meaning, relationship, ordering, time, location, corroboration, or consistency results that are not real.

Accidental or intentional faults

Accidental miss faults are practically impossible to avoid because there are a potentially unlimited number of different analytical methods and processes that could be applied to evidence, any of which might produce something of relevance.

45 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

Accidental make faults are normally the result of inadequate attention to detail, lack of expertise, a non-systematic process, or a lack of thoroughness. These faults are particularly problematic because they produce interpretations that claim things that are not true. The lack of adequate time to thoroughly investigate issues leads to make faults because, in the process of investigation and examination, theories are produced and tested. The human mind tends to make leaps that are the source of human intelligence, but these leaps may or may not be right. A lack of time, care, or expertise, leads to the acceptance of these theories as if they were facts without adequate verification, or their presentation as definitive when they remain somewhat speculative.

Intentional miss faults are commonplace, particularly in adversarial situations. Each side tends to leave out the things that the other side might find helpful to their case and to focus on the issues that best make their own case. Counsel sometimes limits the information available to DFE experts so that they only see the things that tend to aid the client in their case. The DFE expert should be aware that limited information leads to excessive conclusions and take care in drawing conclusions to explicitly state the limits of their conclusions and their basis. If the basis changes, so might the conclusions. Experts who intentionally ignore facts in front of them and draw conclusions that are contradicted by those facts are likely to face serious and justified challenges.

Intentional make faults are almost always fraudulent in nature. Making up evidence or creating conclusions that the expert knows to be false are unethical and in most cases illegal and sanctionable. The DFE expert should seek to identify intentional make faults by verifying results using redundant methods and verifying evidence consistency through analytical methods. Intentional miss faults are often used to cover up intentional make faults. For example, when identifying evidence, such as log files associated with computers that generated other evidence in the case, the party who produces detailed asserted records of one sort but refuses to provide, intentionally destroys, or fails to adequately retain records of related sorts, should be suspected of fabricating the detailed evidence that they proffer. The authenticity of records depends on the collection they are part of, authenticating them typically requires examination

that addresses the context. The DFE expert should identify this issue clearly and assert the potential of spoliation of the detailed evidence provided. If that evidence has internal inconsistencies, the case for intentional spoliation becomes stronger.

False positives and negatives

Faults are important to legal matters when they produce erroneous results or conclusions. The mere presence of an accidental miss does not imply that the expert drew incorrect conclusions or that the evidence doesn't support the matter at hand. In order for a fault to rise to the level of importance that makes it worthy of a legal challenge, that fault should normally produce an error that is material to the case. Even intentional fabrication of evidence doesn't always produce errors that are material. For example, someone who accidentally destroyed a file and created a new version in its place without telling anyone, augmented their accidental miss into an intentional make, but that doesn't mean that the result was inaccurate, only that its pedigree is questionable.

The DFE expert should identify relevant faults, but it is far more important to identify the faults that produce errors and put those errors into the proper legal context. The net effect of faults that are meaningful can be characterized in terms of two kinds of errors; false positives and false negatives.

False positives are results indicating something as true when in fact it is not true. For example, the detection of a condition when the condition was never in fact present, the attribution of an action to a party who did not in fact take that action, or the claim of the presence of contraband when in fact it was not present.

False negatives are results indicating that something was not true when in fact it was true. For example, the failure to detect the presence of a break-in to a computer that was supposed to be reliably storing evidence when claiming that the computer was not broken into, the failure to attribute an action to an actor when it can in fact be attributed reliably based on available information, or the claim of absence of contraband when contraband is in fact present.

In many cases, these sorts of errors are the result of DFE experts making statements that are overly broad, excessively definitive, or

otherwise stated as unilateral and sweeping when they are in fact accurate only for a more limited set of conditions. But in other cases, these are simply the result of process errors in which some key piece of evidence was not properly identified, collected, preserved, etc. or in which something that was not in fact reliable was treated as if it were reliable.

The Legal Process

Legal matters start before any legal filing takes place, and at any time, any system or content might be involved in some aspect of a sequence of events that ultimately leads to a legal matter. As a result, the processes associated with DFE should be part and parcel of every entity's operations at all times. There are defined legal duties to protect and preserve DFE and these have been substantially explored in the literature.[46] The discussion provided herein is based on a loose interpretation of the sequence of events that takes place in legal matters. The actual sequence depends on the specifics of the jurisdiction, the matter at hand, the parties involved, and other case-specific factors.

Figure 2.8 - Legal Processes

Pre-legal records retention and disposition

Before the first paper is filed for a legal proceeding, entities have responsibilities to preserve evidence that could be reasonably anticipated to be involved in litigation. For corporate entities, this entails the creation and operation of a policy and process associated with records retention and disposition. For individuals, the standards are far more lax; however, any situation in which a legal matter is anticipated leads to duties to preserve evidence. The simplest strategy for individuals is to do regular backups of digital information and, if a legal matter seems to be looming, make a copy of everything and put it somewhere safe. For corporate entities and other businesses,

46 "The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production", September 2004 Public Comment Draft.

government entities, or organizations, the issue is far more complicated.

Entities have a responsibility to preserve their records for many legal reasons as well as for reasonable and prudent operations. Some records, such as contracts, publications, historical data associated with patents and other intellectual property, prices charged, and fees paid, are retained for business and legal reasons as evidence of the activities of the entity. Other records, such as records of expenditures and income, are retained for external legal reasons such as government regulations and meeting reporting requirements. Still other internal records, such as electronic mail, memoranda, operating manuals, and notes on when what happened, are retained for internal use, entity long-term memory, and convenience.

Where there is a legal mandate to retain records associated with regulatory bodies, such as tax records, records of controlled substances, employee records, and so forth, entities must retain these records for the legally mandated period, and the entity record retention and disposition process should define these minimum times and identify disposition processes and times after legal limits are reached. Where no such mandate is in place, entities should operate for their own operational efficiency, effectiveness, and convenience, should codify these operational, efficiency, and effectiveness requirements and decisions, and should follow these decisions rigorously. In addition, statute of limitations requirements limit the utility of certain information in certain circumstances, and these statutes should be built into the records retention and disposition process in helping to make decisions about time frames. In all cases, a well-defined retention and disposition process should be in place, operated, and verified in its operation. A legal hold process should also be defined and put in place to assure that prior to disposition of any records that can reasonably be anticipated to be required for any legal proceeding, all legal holds on those records are cleared, and when a legal hold has cause to be in place, appropriate records are preserved and prevented from being disposed of.

Prior to the first filing, and contemporaneous to events of interest, it is important to identify, collect, and assure the proper storage and

handling of any content that might be involved in a legal matter. Perhaps the most important things to do contemporaneously are things that can preserve evidence that tends to change over time or will not exist past a particular time frame. For example, network traffic and voices disappear as they are consumed unless explicit preservation is undertaken at the time they occur. When investigating or acting on digital forensic evidence or matters related thereto, it is often helpful to take notes at the time the activities are undertaken and to retain them as contemporaneous evidence of what took place. Similarly, things like network addresses and host names, network-based lookups, and related information, including versions of software in use and other related configuration information, should be collected contemporaneously because these things tend to change with time, and records of their changes are not uniformly kept. Contemporaneous time and date information, when relevant, performance levels, as measured at the time, and justifications for decisions, as they are made, are best documented contemporaneously.

Digital forensic experts brought in prior to the legal process may be used for a wide range of efforts, including, without limit, internal investigations, preparation for potential legal work, the creation of forensic data collection and examination capabilities, analysis of potential evidence, and so forth. While these may seem like they have a lower standard of care than work during the legal process, the DFE expert should realize that the work they do in preparation may end up questioned at trial, and reasonable and prudent efforts should be applied, proper contemporaneous information should be collected as appropriate to the matter at hand, and all of the elements of the evidence process should be respected, even though no legal action has been filed.

First filing

As of the first filing in a legal matter, a series of events with time limits start to occur. Historical events that apply to the legal matter are limited by statute of limitations limits depending on the nature of the charges and specifications and the jurisdictions that apply. The Constitution of the United States,[47] as well as many other similar

47 The Constitution of the United States of America, which can be viewed at http://www.archives.gov/exhibits/charters/constitution.html

legal mandates from other jurisdictions, requires (in the 6th amendment) "In all criminal prosecutions, the accused shall enjoy the right to a speedy and public trial,...". The right to a timely trial means that from the first legal filing to the start of the trial must be speedy. But beyond this, courts set calendars and require that they be met. Late filings result in adverse rulings, and as a result, there is often a rush in the legal system for those who are working on issues related to evidence.

In most legal matters, before the force of legal process can be used to secure and process evidence, a legal action must be filed. For example, before a subpoena can be issued, a lawsuit normally has to be filed. The first filing then triggers notice and preservation requirements and allows legal papers to be filed to compel actions on parties.

Notice

Notice is given of various things during the legal process, starting with notice of the existence of a legal action. Various sorts of non-disclosure, confidentiality, work product, documentation, and other sorts of requirements are given in various forms throughout the legal process. Because the legal environment tends to be relatively unforgiving of those who fail to comply with judicial orders and similar things, it is important to respect all of the notices given and to communicate all such notices with appropriate legal staff in a timely fashion. In the case of an entity that is given notice of a legal matter, it is important to start the legal hold process within the data retention and disposition process, and to immediately and accurately identify, collect, and preserve all relevant evidence. Once notice is given, there is a duty to preserve evidence.

Preservation orders

In many cases, preservation orders are given with respect to evidence. It is important to get timely preservation orders in order to assure that critical evidence is not lost. The DFE expert is often called upon to assist the legal team in identifying the sources and nature of evidence that should be sought, and this is often codified in preservation orders and the language of demands for evidence. Timeliness requirements stem largely from the data retention and disposition issues related to different entities. For example, many

Internet Service Providers (ISPs) only retain records for periods of days to weeks, and in some cases, intentionally avoid retaining records to facilitate anonymity for their clients. Jurisdictions sometimes mandate preservation of particular data, like calling information not including the content of calls, as part of their national security or other legal mechanisms, but gaining access to this sort of data requires effort on the part of the legal team, and the costs of such actions may exceed the value they bring to the legal matter. Courts often rule, particularly in civil matters, that the value of the evidence in terms of its probative utility is exceeded by the cost of production, and this effectively limits the preservation and production process in some cases.

Disclosures and productions

Documents are typically produced either as part of disclosures made by the parties or as productions in response to legally authorized demands by parties. These productions and disclosures constitute the bulk of the digital forensic evidence in most cases, but they also include information that brings context to the evidence, including the claims being made, assertions by the parties, and the basis for those claims and assertions. Examination of the evidence should yield results that are consistent with truthful disclosures. When there are inconsistencies, or when the basis is not adequate to support the contentions made in the claims or disclosures, the digital forensics expert is typically tasked with identifying and clarifying such inconsistencies and lack of basis, and the results of these efforts form the basis for effective challenges to the evidence and the legal case.

Disclosures and productions are often applied tactically by the parties to make their case while preventing challenges. For example, it is fairly common for parties to disclose printed copies of digital information but not offer the digital forensic evidence. In such a case, it is the responsibility of the other side to demand original writing in digital form so it can be forensically analyzed. Large volumes of data are sometimes provided and select data contained within those large volumes may contain the key information required to understand what took place. It is the responsibility of the party receiving such volumes of data to go through it all and, when that data indicates the presence of other systems or content, to

identify those systems and content for further demands of disclosure.

To the extent that a disclosing party intentionally subverts the process and intentionally creates high levels of effort by the other party without basis, it is sometimes possible to get sanctions against the offending party, particularly when the aggrieved party can show that the other side knowingly and intentionally misled. The DFE expert that identifies such instances and helps to bring about those sanctions is bringing added value to their side of the case because the other party may have to pay for the cost of much of the legal effort and the fees of the expert in analyzing materials that were needlessly produced when they were known to be irrelevant, or productions that were contrary to the judicial orders in the matter. The DFE expert will often write a report on a legal matter and this report will be disclosed to the other parties at some point in time. The reader is advised to review other sources for more details.

Depositions

Depositions are testimony given with lawyers present and a legal recording made of the proceedings. The questions are typically asked by the other side, and the answers are sworn testimony that bears all of the same requirements of testimony in open court. Witnesses, including experts, are typically deposed prior to trial so that the attorneys can gain valuable information related to the matter at hand and to which they have a right. The right to face one's accuser[48] (the fifth amendment) includes the right to question them and any and all witnesses that may be brought. This means that the DFE expert who will ultimately write a report or testify in open court will be deposed and that the DFE expert may be asked to offer assistance to lawyers who will be deposing the opposition when the issues relate to DFE.

DFE experts brought in to help lawyers prepare for depositions have a somewhat different role. For example, they may help to identify and prepare items of evidence that will be used in questioning a witness. They may help the legal team identify the

48 The Constitution of the United States of America, which can be viewed at http://www.archives.gov/exhibits/charters/constitution.html

proper sequence in which to present questions in order to make a series of legal points and provide specific items of evidence that allows those questions to be pursued one after the other. For example, to get a witness to admit that they don't know how a process used to develop evidence actually took place, they might provide an example for the lawyer to show the witness with a set of specific questions related to the piece of evidence. Depending on the answers given, different following items of evidence might be presented that show that the answers given were not correct. The witness may end up contradicting themselves, or admitting the limits of their knowledge of the facts in the case, and this might result in the evidence and the witness losing their credibility. Of course the same may be done by the opposition, and that's why the DFE has to understand these issues even if they are not being asked to help the lawyers prepare for a particular witness.

As the subject of depositions, the DFE expert has a legal obligation to tell the truth, and of course failure to do so may result in enormous problems and legal implications for the expert. But this is only the beginning of the issues that the expert faces. Great care should be taken in answering questions and great precision should be sought in the application of those answers. In many cases, experts answer too quickly, interrupt the questioner, don't answer fully, answer things that were not asked, and make other similar mistakes.[49] Preparation for depositions should be undertaken with the lawyers in the case, and it is always advisable to do a practice deposition the day before the real one to reduce the stress and get a sense of the sorts of questions that will be asked in the particular case and to make certain that the answers are precise, accurate, and address the questions. The DFE expert should think through the totality of issues involved in the matter and recognize the limits of what they may be able to testify about as well as the features so that they are prepared for the potential sequences of evidence and questions they may be asked.

Motions, Sanctions, and Admissibility

Motions in legal matters are often accompanied by expert reports relating to the evidence, and when the evidence in question is

49 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

digital in nature, the DFE expert will likely end up writing those reports, or at least signing off on declarations written by lawyers. It is vitally important that all such declarations and reports in support of motions or used in legal matters be carefully written and as precise and accurate as the expert can make them. While most non-legal environments instill a sense of coming to consensus and writing an agreeable work product that others will like or buy into, in the legal environment, and particularly in support of motions, it is the precision and accuracy of the product that matters. In such a situation, the DFE expert is writing an opinion based on facts and properly applying a scientific methodology. The DFE expert is the final authority on such a report and must not be convinced by others to say things that they do not truly believe to be the case or things that they do not believe can be demonstrated by the proper application of scientific methodology to evidence in the case.

Typically, the results of such writings are "facts" asserted to be true by the side proffering them. The other side has an opportunity to dispute these facts, but if they are undisputed, they become legal facts for the case, and as such, constitute the basis for the trier of fact to make a judgment. If they are disputed, the other side had better have an expert who also has a scientifically based methodological approach that, using the same evidence, shows that the things one expert asserts as fact are not in fact true. This direct sort of difference of opinion is relatively rare when properly qualified experts testify in legal matters, and in the case of DFE, it is almost never the case that the experts disagree on the bits. Almost all interpretation of the bits in the DFE arena are testable, and the other side may well test them as the DFE expert may be asked to test them when presented by the other side.

Motions can also result in the exclusion of evidence that may be vital to a case, limits on the interoperation of evidence, the removal of an expert from a case, or any of a wide range of other outcomes, including the end of the proceedings and termination of the case. Motions are used to get sanctions, limit admissibility, and for essentially all other aspects of a legal matter.

Pre-trial

In addition to motions and other legal maneuvering, before trial, DFE must be analyzed, interpreted, attributed, sometimes reconstructed, and prepared for presentation. This includes the preparation of reports, exhibits, and demonstrations, preparation for testimony, and assistance in challenging the testimony of others.

Report preparation consists largely of describing the context of the report and the background of the individual preparing it, the processes and tools used related to the evidence at hand, the interpretation and attribution of the evidence in light of the case, and expert opinions related to the evidence and the context of the case. Depending on the specifics in the matter and the interests and requirements of the legal situation, the report may contain many citations and attachments. In some cases, very short reports are provided, and many lawyers believe that judges will not read more than a few pages of an expert report, but some cases call for a great deal of detail, cover hundreds of thousands of claimed items of evidence, and involve many complex issues.

Preparation of exhibits that support expert opinions have to be accepted by the court and meet standards of admissibility, including being reviewed by the other parties to the case and challenged for all of the factors involved in admissibility. Complex areas of digital forensics may include a short tutorial given to the trier of fact on the underlying operation of the systems involved, such as a depiction of what an IP datagram consists of and how a particular protocol works, with examples provided that are relevant and that demonstrate the issues in the case. Demonstrations, such as a live session where an email is sent using manual entry of the protocol elements, it is received by a receiving computer, and the logs and output generated are shown to the jury are far less common than written reports with examples demonstrating these activities and assertions that these accurately represent the events that transpired. This is not only because live demonstrations are less reliable than pre-recorded ones, but also because these sorts of reconstructions are sometimes more prejudicial than probative, take a lot of time, and are rarely important enough to the legal matter to justify their use. They are also subject to challenges and live counter-demonstrations, and are thus problematic. The most

common type of evidence shown to a jury is a computer printout or a large chart that is prepared before the trial and used to bring clarity to the trier of fact. Increasingly, courts are using video displays to show these sorts of charts and other similar evidence, and these technical means of presentation have to be prepared, shown to the opposition, and presented as evidence supported by expert testimony.

Notes, draft reports, emails, FAXes, and other similar supporting records of acts are often subject to discovery by the other side. As a result, in the pre-trial phase, it is important to use special care in handling and creating these materials. In many cases, counsel makes the requirements for such handling clear in advance of the work by the expert. But in all cases, the well prepared expert should anticipate the needs of handling for DFE and have systems and processes in place to avoid the pitfalls before falling into them.[50]

Testimony

The expert or lay witness who presents digital forensic evidence in front of the triers of fact normally does so live and in person. The members of the jury or the judge trying the case are typically sitting within a few feet of the witness who is asked specific questions similar to those given in a deposition. Evidence is brought up in front of the court and is readily visible to the witness and trier of fact as the expert explains what it is, how it came to be, how it is interpreted, and what it means. Cross-examination allows other parties to ask questions about the evidence and the opinions, and to identify inconsistencies between what is said at trial and what was said in reports and depositions.

Most judges and juries do not have expertise in computers, programming, electronics, or other aspects of DFE, just as they usually know little about the chemistry of DNA or the fluid dynamics of blood as it splatters. As a result, the expert witness is tasked with educating the trier of fact about the underlying facts and the nature of the systems that create, process, store, communicate, and present the DFE. For this reason, the expert usually has a lot of explaining to do, and much of it is about things that most experts find to be rudimentary. However, this explaining lays the foundation

50 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

for the detailed conclusions and opinions that the expert gives and that make the difference in the case, and it must be accurate and precise, while still explaining the issues to people who don't know much about the subject. As such, it is a challenge.

This explanation of detailed scientific methodology and its proper application applies to each and every step of the process associated with the evidence, and each of those steps may be challenged by the other parties to the case. It is vital that the expert testifying about such evidence be able to explain why they have the opinions they have, how they came to those opinions, and at a detailed level, the mechanisms that cause the opinion they give to be correct. Legal cases have turned on experts who were or were not able to explain the operation of the file system from which they collected DFE and how that file system is used by the low-level system calls within the operating system on the computer that was examined. It is all too easy to answer questions in such a way that they are easily challenged, to assert knowledge that is not really clear, to become sloppy and make guesses, to make a miscalculation, or to make other sorts of errors, particularly when answering complex questions in real-time in front of strangers.

Case closed

After all of the other aspects of a case are done, regardless of who wins or loses, the DFE often has to be disposed of in keeping with court orders. Legal matters rarely require that the evidence be destroyed using techniques that are difficult to apply, but it is common that confidential information must be removed using reasonably sound techniques so as to assure that it is no longer available to the expert or anyone else. This includes backup copies, data collected by internal search mechanisms, cached copies, copies on paper, tape, and other media, and residing on all affected systems and peripherals. For this reason, it is useful for the DFE expert to use special precautions when originating, examining, and storing matters related to legal cases so that the back-end process does not become complicated or overly burdensome. While it is prudent to keep backups, it also implies the need to remove copies from those backups.

Duties

While duties have been discussed previously, it is worth the effort to reiterate the major duties identified for digital forensic evidence with regard to experts and entities.

Honesty, Integrity, and Due Care

While it may seem obvious, those working in the digital forensics field have special requirements for honesty, integrity, and diligence in their work. Above and beyond the normal level of care seen in common use, those working in legal settings really should meet a higher standard.

Previous writings, public statements, legal proceedings, and other records of past performance are all subject to challenge in legal settings, as long as they are relevant to the issues in the case, which in the case of an expert witness, includes their credibility as an independent expert in the subject at hand. The Internet and other digital fora and media produce a great deal of history that may come into play in legal settings, and the expert in DFE is most likely to have a lot of such information about them readily available on the Internet because that's where much of the work in their field is done. A search of a well known person who has done a career worth of work using the Internet can easily yield hundreds of thousands of pages of material, and not all of it will be factually accurate, but it is all available to be used in challenges to the credibility of the witness.

The challenge of due care is far more daunting in that there are really no well established standards of care associated with information and information technology, despite the common use of the term "best practice". There is a lot of misinformation in the world, and the DFE expert who relies on information from sources that are less than credible may lose their own credibility by believing them without taking the proper precautions in evaluating what they assert. The use of non-authoritative sources, such as online encyclopedias that are created by the Internet community, while useful in everyday applications, may not be up to the standards required for a legal proceeding, and if they are used as sources without proper verification, they may end up destroying the credibility of both the case and the witness in the process.

A diligent effort in a legal setting typically means relying predominantly on things that the witness has personal knowledge of. For example, in validating a time and date, lacking any other basis for its validity, the DFE expert should do some testing or seek out some independent evidence that supports the claims being made. The "take it on faith" approach is problematic when the issue is important to the case. On the other hand, legal counsel in a case may direct the expert to only attend to certain issues, and in these cases, the expert cannot realistically refuse to do what they are being hired to do. The solution typically comes in being diligent in how information is presented and in how questions are answered. If independent validation was not undertaken, the results should be stated with appropriate caveats, even if that presentation may make it seem "legalistic". It is, after all, a legal matter.

Competence

Professional societies like the IEEE have codes of ethics that are worthy of particular attention to those engaged in working on DFE. In particular, the IEEE code of ethics insists that member agree "... 6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations". In the digital computing arena, as in many other businesses, there is a history of successful individuals exaggerating their backgrounds or qualifications in order to make progress in their careers. But in working on legal issues, this is problematic for all concerned. It is incumbent on anyone working in this field to recognize what they do and do not know and to limit their work and testimony to areas in which they are professionally competent to do the work they are doing. In addition, to the extent that the potential expert is not comfortable with their knowledge of the particular issues in a case, they have a duty to their clients as well as the courts to identify their limitations to counsel. To the extent that the expert can gain additional competence, knowledge, and experience in a specific subfield through diligent effort in a very short time frame, this is certainly something worth doing, but the expert who is not adequately knowledgeable is risking the well being of their client on their ability to learn quickly, and to do so without notice is unethical.

Retention and disposition

There are specific legal duties associated with retention and disposition of DFE and other materials related to digital forensic matters. The pre-legal requirements are largely described above under the "Legal Process" section above in the "Pre-legal" subsection, and the post-legal requirements are discussed briefly in the "Disposition" subsection of that same section. The interested reader should read the Sedona Conference report[51] thoroughly and look for updates as they become available.

The science of digital forensic evidence examination

Digital forensic evidence examination, if it is to be effective in a legal setting, must be a scientific activity. But what constitutes the science of DFE examination?

As a baseline, the student of DFE examination should certainly be aware of the issues of diplomatics, which are only today being translated into the digital arena, and the underpinnings of electrical and computer engineering and computer science. Without these as background, trying to understand the science of DFE examination would be, as it has been for many, an eternal effort to make incremental improvements with occasional minor breakthroughs, largely in going where others have gone before. With these as context, the student has a solid starting point and can leverage the thousands of years of work of others to advantage and move more quickly into information age science.

The principles of scientific inquiry are, to some limited extent, debatable, but overall, they consist of four basic elements:

- Studying the past, understanding current scientific theories, methods, and the experimental basis for believing the theories, and understanding the limits of current science and how it can reasonably be questioned, tested, and refuted. This is the study of diplomatics, electrical and computer engineering, computational science, and related areas, and keeping up with current literature in the field.

51 "The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production", September 2004 Public Comment Draft.

- Performing analytical processes associated with the science, identifying inconsistencies that are not readily explained by the limits of the methodology or the current scientific theory, generating reasonable hypotheses about the reasons for inconsistencies, and performing analysis to determine whether those hypotheses are indeed reasonable and what they predict that the current theory does not predict correctly.
- Proposing experiments that allow scientific theories to be tested in ways that could generate refutations if the theories are not correct and confirmations if they are correct, predicting the outcomes of those experiments that would confirm or refute current and proposed hypotheses, performing those experiments using methods that are adequate to accurately test the hypotheses in question, and properly characterizing the results of those experiments so as to confirm or refute the rival theories.
- Properly documenting the results of the experiments in a way that allows them to be independently repeated and confirmed or refuted, and interpreting the results of those experiments properly within the realm of both the old and new scientific theories.

In the DFE examination context of this book, these four elements of science are characterized.

Understanding the current scientific theories and performing analytical processes using those theories while recognizing their limitations is largely covered under the subjects of analysis and interpretation, with interpretation also being part of the proper documentation of experimental results.

Attribution is largely about associating causal mechanisms with traces, which is, in essence, a specialized case of analysis adding in the creation of additional hypotheses and historical background and research.

Reconstruction is the experimental part of digital forensic evidence examination, and it bears the burden of doing tests that are not part of the well-defined pre-existing body of knowledge associated with digital systems. As such, it is also

the most definitive process in clarifying the difference between reality and theory.

Other resources

There are many books that describe digital forensics techniques, particularly in the area of the use of specific tools and the aspects of identification, collection, analysis, and attribution. But there are far fewer books that deal with the issues of interpretation and none on reconstruction.

There are some conferences in the digital forensics area, such as the "IFIP Working Group 11.9 International Conference on Digital Forensics",[52] tracks within other conferences, such as the "Hawaiian International Conference on System Sciences", emerging refereed journals, such as the "Journal on Computer Crime", and some books suitable for use in graduate courses.[53,54, 55] Another excellent source of practical information is the High Technology Crime Investigation Association (HTCIA.org).

However, as a field, digital forensics is still young, and much of the current technical effort largely ignores the legal aspects of the field.

The reader is also encouraged to go beyond the coverage of this book in terms of diplomatics, computer engineering, and computational sciences and to research these fields, starting with the references we provide and your local university. The Internet is, of course, a rich source of quality information, but it is also a good source of misinformation. By searching for writings by authors referenced, you are more likely to find higher quality information and learn to evaluate other information you find.

Questions

1. Given the characterization of examination provided in this overview, what other subspecialties might reasonably be

52 "Advances in Digital Forensics II", 364 pages, Springer; August 30, 2006, ISBN-13: 978-0387368900

53 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

54 T. Johnson, Ed. "Forensic Computer Crime Investigation", Taylor and Francis, 2006.

55 E. Casey, "Digital Evidence and Computer Crime, Second Edition", 688 pages, Academic Press, March 8, 2004, ISBN 0121631048

identified, and how might the overall digital forensics problem then be characterized?

2. How should an examiner deal with the legal context in terms of making decisions about what to pursue and what not to pursue in the examination process? How do the calendar, costs, and strategies impact the examiner? How does the theory of the case interact with the application of the methodology of examination? How do jurisdictional issues, case type, and standard of proof impact the examination process?
3. What processes are used today with respect to DFE tools and how does the examiner use these processes to assure that the results of examination are suited to the legal requirements?
4. What sort of qualifications do you as an examiner have to do the sort of examination discussed herein, and what sorts of examinations would be within and outside of your area of expertise?
5. Given the wide range of possible challenges to DFE and to the examiner, how can you provide reports and testimony that will hold up to the most stringent scrutiny?
6. Since the legal process provides potential uses of examiner expertise at all phases, how can the examiner get involved in the process at different stages to improve the quality of the evidence and its use in the legal context?
7. Knowing something about admissibility of evidence, how does the examination process go to the issues of admissibility, and how can an examination cause what might otherwise be considered valid evidence to be seen as invalid? How could an examiner faced with potentially invalidatable evidence help to rehabilitate that evidence through an examination process?
8. Taking the overall context of digital forensics and the somewhat more narrow scope of the DFE examiner in this process, is a specialized subfield for examination warranted? If so why? If not, why not?

3 The physics of digital information

Just as the physical world has physical properties that characterize how things work and what can and cannot reasonably be expected to happen in given circumstances, so does the artificial world of digital systems. Understanding the physics of digital information in the artificial digital world of bits and machines that operate on them is fundamental to the examination of digital forensic evidence because it allows the examiner to characterize what can and cannot reasonably be expected to happen in a given circumstance within that digital artifice.

Causality, measurement, precision, and accuracy

A basic assumption of science is causality. Causality asserts that cause (C) acts through mechanism (m) to produce effect (E). This may be written as $C \rightarrow^{m} E$.

In general, measurement is limited in both precision and accuracy. Thus we are only able to measure cause, mechanism, and effect to within some bounds. These bounds limit our ability to be definitive and thus effect how crisply we can locate and characterize things. In the analog world, this is reflected in bounds on measurements. For example, we might measure a distance as 23.5 meters, but because our measurement mechanism (e.g., we view a tape measure) is imperfect, results are necessarily imperfect. The error rates and amounts of such measurements are then vital to understanding the result. For example, we might indicate that the distance was 23.5±0.1 meters to show that the results may be off by (plus or minus) 0.1 meters.

Note that there is a relationship between the precision shown for the measurement (3 digits, one after the decimal point) and the accuracy (0.1 meters). While the inaccuracy (size of the error) may be greater than the precision (smallest grain of difference) of the measurement, a measurement cannot be more accurate than it is precise. Thus, if the measurement was accurate to ±0.01 meters, the precision of the measurement had better be to two digits to the right of the decimal point (e.g,, 23.52±0.01). While, precision can exceed accuracy (e.g., 23.52±2), the excess precision is then not informative and may be misleading.

The nature of digital forensic evidence

Digital forensic evidence has some common, even if not universal, properties that are worth considering before exploring frameworks and models used for analysis.

The physics of DFE is different from that of matter and energy

The physics of matter and energy, in general, is not the same as the physics of digital information. While all digital information ultimately resides in some physical form that is subject to the laws of physics, the subject matter of this book explicitly excludes the devices and mechanisms used to store and transport the DFE. Some of these differences are rather substantial in terms of the way it may impact the thinking about evidence in legal matters.

Finite granularity

DFE has finite granularity, with the smallest grain of space being the "bit", or binary digit; and the smallest grain of time being the fastest clock cycle of the underlying digital mechanism.

Unlike matter which may decompose into molecules, then atoms, then particles, and who knows what of even smaller size, DFE is composed of bits. Bits are digital things that have one of two distinct and clearly discernible values. Whether we represent these values as 1 and 0, True and False, T and F, or using any other representation, there are two and only two states that any atomic particle of DFE can be in. This has many substantive implications to examination.

For example, given any real system, there are only a finite number of different possibilities of the settings of bits, because there are only a finite number of bits. Thus the maximum precision of results is also finite. If we know how many bits are involved and the representation being used, then the maximum precision can also be known. Further, for digital mechanisms, precision and accuracy can always match with no residual error. A bit can be measured as 1 ± 0 or 0 ± 0 with no measurement error and perfect precision.

A result presented as more precise than is theoretically possible may be characterized as misleading. But in the digital arena, measurements can be perfectly precisely and accurate. Of course this doesn't mean that results are properly presented to reflect this.

Time in the digital world is also finite granularity. In digital systems, synchronization is used as a fundamental method of avoiding race conditions[56] that produce unpredictable results. The time granularity of a system is often expressed in clock or instruction cycle times or rates. A gigahertz processor processes some level of instruction at the rate of 10^9 per second. Internally, the mechanisms may operate at even higher speed, and some self-timed or continuous processes may take place, but at the end of the day, a finite clock cycle is necessary with limited granularity in order to produce the digital world that emerges from digital system designs of today.

But even if the clocks were not this way, there is a finite granularity to the representation of time by any digital mechanism because there are only a finite number of bits available in total. However many bits that is, and regardless of the representation used, a finite number of bits can represent no more different times than the number of different states of that set of bits. To be specific, given n bits, the total number of distinct values that can be represented is no more than 2^n because that is the total number of distinct settings of n bits. For example, with 2 bits (n=2), the only 4 possible values for the 2-bit sequence are {(0,0), (0,1), (1,0), (1,1)}. While the same n bits may represent different things in different situations, for any given representation, there are no more than 2^n values for n bits.

Observation without alteration

In the physics of the natural world, it is generally considered impossible to observe a particle without altering its state. But this is not true of bits. Because the physical mechanisms used to represent bits are composed of many particles, contain a great deal of internal redundancy, and in the case of electronic storage, are regenerative within defined bounds when slightly altered (e.g., the voltage may change slightly but there is a feedback mechanisms that restores it to its previous value within defined tolerances), bits can almost always be observed without changing their physical state enough to change the bit being represented. While physical mechanisms that represent a bit may change during observation, these changes are, by design, not large enough to change the represented bit.

56 A. Friedman and P. Menon, "Theory and Design of Switching Circuits", Compuer Science Press, Woodland Hills, CA, 1975.

By way of example, you can see the one-zero-one that follows (101). Since you have read the digits and they remain unaltered (as you can verify by reading them again and again), in terms of the digital information they represent, they are observed and unaltered. While at some low level of physicality there may be alterations in the paper and ink as a result of your observations, at the level of the representation of the bits on printed paper, they are unaltered.

Exact copies without altering the originals

One of the implications of bits as the lowest granularity of DFE and observation without alteration is that exact copies can be made without altering the original. While in the physical arena this is not true, in that any experiment or interaction with matter causes possible and externally indiscernible alterations to the state of that matter, in terms of digital values, experiments, analysis, duplicates, changes of media, form, and format may all be done without altering the digital values associated with the sequences of bits constituting the evidence in any way. By way of example:

10001011100111010011101001010111111010101010

10001011100111010011101001010111111010101010

In this case, the original duplication was made with a copy and paste, but by the time you read it, it will have also been made by a variety of electronic processes, including transforms into different formats and over networks as well as printing.

As this example shows, the physical vs. digital worlds are different in this way. The redundancy in the various processes that ultimately produce the printed representations that you can read lead to an “exact copy” that did not “alter the original”. Each of the lines of binary digits is printed with different ink on different parts of paper, and close inspection will find that they are different at small granularity levels. And yet the net result, you can see for yourself, is that the copy was made, and the original and duplicate are identical, not in every way, but at the finite granularity of the bits. These are exact copies of the digital information, even though they are not exact copies of the physical realization.

While it is possible to alter DFE, proper technique and care in handling may assure that duplicates of digital evidence is exactly

what it was when collected. Copies are, for all purposes related to the digital nature of the evidence, equivalent in every meaningful way. This means, among other things, that there is no such thing as a unique piece of DFE that ties that evidence definitively to a particular physical thing. The notion of a "match" is precisely definable, and yet a match between two sequences of bits does not generally mean that they came from the same source, or anything else of the sort. They may be produced by many systems or finite state machines, any of which may present.

Effect does not imply unique cause!

This may be written as $\overline{E \rightarrow C}$, which is to say, effect does not imply cause. In essence, there may be many mechanisms (m_1, m_2, …) and associated causes (C_1, C_2, …) such that ($C_1 \rightarrow^{m_1} E$, $C_2 \rightarrow^{m_2} E$,..).

You can "take" bits without removing the original

In the physical arena, when you take something of value, it is no longer available to its previous owner. But in the digital arena, you can make an exact copy and take it away without removing the original. Therefore, the owner of the original still has what they had before, even if they no longer have exclusive control over it. This means that "theft" need not deprive the original owner of all of the utility of what was stolen, even though it may alter property rights or value in markets. Possession does not imply prior contact, as it might in the physical world, and the notion of "transfer"[57] [58] from classic evidence discussions, is not applicable, in that there is no division of a bit into parts that transfer to other bits.

Bits can move very - but finitely - quickly from place to place

The movement of bits from place to place can happen very quickly. In fact, bits can move quite literally at the speed of light, even though they actually move quite a bit slower in most cases. But just because they can move quickly, doesn't mean that they move

57 K. Inman and N. Rudin, "Principles and practices of criminalistics: the profession of forensic science", ISBN# 0-8493-9127-4, CRC Press, 2001

58 E. Locard, "The Analysis of Dust Traces", Revue International de Criminalistique I. #s 4-5, 1929, pp 176-249, (translated into English and reprinted in 3 parts in A, J. Police Science, 1930 in V1#3, May-Jun 1930, pp276-298, V1#4 Jul-Aug 1930, pp 401-418, and V1#5 Sep-Oct 1930, pp 496-514.)

instantly, or that they violate the laws of time and space we associate with other things. More specifically, causality still applies in that if A causes B, then A must happen before B, and if B happens before A, then A cannot cause B. Digital systems have known, verifiable, and testable time, rate, and other performance characteristics that can sometimes be used in examination to identify and determine possibilities and impossibilities.

DFE is created by artificial means

DFE comes in the form of sets of sequences of bits. These bits are normally artificially generated by automated mechanisms, and such mechanisms produce characteristic sequences based on the way they are designed, implemented, and operated. For the most part, the designers of these systems use specific syntactic methods to store and communicate information so that it can be easily read by other mechanisms for automated processing, display, and other uses. The “other” processing mechanisms also have characteristics associated with the way they were designed. For example, a camera might have anti-aliasing and motion stabilization algorithms that produce characteristic features in outputs associated with certain types of edges or motions. Many such mechanisms place explicit marking information (e.g., headers) at the start of and/or between subsequences to allow the other mechanisms that use the results to readily parse the content for anticipated uses.

Finite state machines are the most common artifice

The mechanisms that produce DFE are typically deterministic finite state machines (FSMs) with internal states, transform mechanisms, input mechanisms, and output mechanisms. FSMs take inputs from a finite alphabet, and based on their current state, produce next states and outputs from finite sets of states and output symbol sets.

This is typically represented as M:(I, O, S, IxS→S', IxS→O) where M is the FSM, I and O are the finite input and output symbol sets (alphabets), S is the finite current state (a set of bits), and S' is the finite next state taken from the same symbol set as S.[59] [60] The finite

59 E. F. Moore, “Gedanken experiments on sequential machines,” Automata Studies. Princeton, N. J.: Princeton University Press, 1956, pp. 129-153.

60 G. Mealy, "A Method for Synthesizing Sequential Circuits". Bell Systems Technical Journal 34: 1045–1079, 1955.

alphabets have known total sizes, such that for an alphabet of k elements, only k possible values exist. These alphabets are implemented in the digital space by sequences of bits and, given that the alphabet can be represented by n bits, the total number of symbols in the alphabet cannot exceed 2^n.

The total number of inputs, outputs, and states are therefore limited to $2^{|I|}$, $2^{|O|}$, and $2^{|S|}$ respectively, where the |x| operator indicates the size of the set x, in bits.

Time transforms the artifice

As FSMs operate over time, sequences of inputs appear in the form of symbols from the alphabet I. The IxS→S' and IxS→O transforms produce new values for the state and output, respectively. There is normally a clocking mechanism associated with the implementation of the FSM that allows the system to reach stability and retain states for periods of time. During these time periods the physical mechanisms change values to those of the next values to appear at the input, and retain states to allow outputs to be provided to other FSMs as their inputs. As the clock allows and disallows changes, FSMs produce sequences of states and outputs that can be described by finite time granularity state and output sequences.

For example, we define FSM_+ with 1 bit of input (I={0,1}), one bit of output, (O={0,1}), and one bit of state (S={0,1}), with transforms as shown at right. If FSM_+ has initial state S_0=0 and input sequence (011101), it produces the state sequence (011101) and output sequence (111101). Note that the initial state is not indicated in the state sequence shown.

SxI	S'	O'
0,0	0	1
0,1	1	1
1,0	0	0
1,1	1	1

FSM_+ transforms

Thus digital systems have causality, and FSMs are the mechanisms by which cause is transformed into effect. ($C \rightarrow^{FSM} E$). These causal systems act precisely in the way described by the mechanisms, and we can understand the mechanisms with complete precision and accuracy in the digital sense.

Current state does not always imply unique history

FSMs have many properties that have been explored in computer science, computer engineering, mathematics, information science,

and information protection literature. One such property is that any FSM that has more bits of input sequence than bits of internal state, must have repetitive states. Thus all states and subsequent outputs of such an FSM are not unique to a unique sequence of prior inputs. For example, FSM_+ has 1 bit of state, so as soon as it has at least 2 bits of input (2 steps of the FSM), it is guaranteed that it will repeat either a 1 or 0 state. To demonstrate, S_0=0. I_1=0 →S_1=0 (repeated). I_1=1→S_1=1 (unique). But since all state values {0,1} have now occurred, S_0 and S_1 unique → S_2 not unique. How FSM_+ got to S_2 (i.e., (I_1,I_2)) cannot be uniquely determined from subsequent outputs and states. Here are all of the possibilities:

S_2=0	S_2=1
S_0=0, I={(0,0), (1,0)}	S_0=0, I={(0,1), (1,1)}
S_0=1, I={(0,0), (1,0)}	S_0=1, I={(0,1), (1,1)}

This example shows that given only S_2, and FSM_* details, we cannot tell what prior states or sequences occurred. As FSMs go from state to state with time, the number of different input and state sequences potentially producing current state and output grows. The mechanisms and effects are identical, but two different causes cannot be differentiated from effects. ($C_1 \rightarrow {}^{m}E$, $C_2 \rightarrow {}^{m}E$)

But there are also FSMs that partition state and output sequences based on previous input states and sequences. For these FSMs, prior state and input sequences may be partitioned, even if not always uniquely, into class sets.

Homing sequences and FSMs

For certain classes of FSMs, the initial state may be reachable after the FSM starts executing. A sequence of inputs that produces the initial state is called a "homing" sequence. As an example, FSM_+ has a homing sequence (0) in that any 0 input will produce state 0 as the next state.

Alternatively many FSMs may have a state that is sufficiently similar to the initial state that all available outputs at a later time are identical given that subsequent inputs are identical. This is common in FSMs that “initialize” as they start up and never repeat the initial state during normal operation. For the purposes of discussion, we

will call an input sequence that produces such a state a "partial homing" sequence.

For some FSMs, there is a homing sequence that works regardless of the current state of the FSM, while for other FSMs, there may be partial or complete homing sequences that only work in subsets of the FSM states. It is guaranteed that, in theory, there exist infinitely many partial homing sequences for any FSM that can receive unlimited input, if only by repetition of a single such sequence.

FSMs may also have submachines with these properties so that once the FSM enters a submachine it never leaves it, or leaves it only under certain limited conditions. For example, for an FSM that starts in state “0” and goes to state “1” forever as soon as a “1” input arrives, the entire input history is known if the state is “0”.

Traces of FSM execution

We will call a sequence of inputs, states, and/or outputs set into an examinable form in the course of activity of an FSM and as a byproduct of that activity, a “trace” of the execution of the FSM. A trace of the entire sequence of inputs, states, and outputs of an FSM is called “complete”, a trace that is not complete is called “partial”, a partial trace from which a unique complete trace is reconstructable is “equivalent” to a complete trace, and a trace that is not complete or equivalent to complete is “incomplete”.

A complete trace provides the entire history of an FSM, but clearly, a non-trivial FSM that has executed even a single state change cannot store a complete trace of its own state, because it requires the entire state of the FSM to store its current state. Therefore, a complete trace of at least one execution step can only be created and stored by an external mechanism.

Given a known initial state and finite execution sequence, an FSM might be designed to produce states that retain its input sequence. Since an FSM must always produce the same state and output sequence for the same initial state and input sequence, such an FSM would contain a partial trace equivalent to a complete trace. Similarly, a mechanism that could store an initial state and input sequence of another FSM could store a partial trace equivalent to a complete trace.

In practice, the mechanisms that implement digital systems today do not store traces of the execution of the FSMs that, for example, implement the central processors, memory, input and output, and other component mechanisms and their internal workings. As a result, for digital forensics, examiners essentially always work exclusively with incomplete traces.

How time transforms the artifice

FSMs have well defined characteristics and are very predictable in how they move forward in time. But because they produce only incomplete traces and many prior states and input sequences produce identical later state and output sequences, there are two important rules that are almost always true for the DFE examiner:

Given initial state and inputs, later outputs and states are known

Given final state and output, inputs and prior states are not unique

We can always drive an FSM "forward" in time through sequences of inputs from an initial state to see what it will do. Thus we can perform repeatable experiments in the forward direction of time with identical results. But in general, current state does not imply unique history, and with only an incomplete trace, we cannot reverse time in a digital system and get a unique causal sequence or prior situation. Stated differently, many causes may lead to the same effect, but any given cause always leads to the same effect.

The digital artifice over time is, in general, a many-to-one transform. Furthermore, inverting time in an FSM produces potentially enormous class sets of possible prior states and inputs, and determining them precisely is too complex to be done for nontrivial systems.[61] This is at odds with the current model of the natural world, in that physical space is generally believed to have an essentially infinite number of possible states and to increase in entropy over time so that order is always reduced. Thus:

Digital space converges while physical space diverges with time.

This is a very important concept to understand in that the current assumption underlying the physical space that we live in is that

61 M Backes, B. Kopf, and A. Rybalchenko, "Automatic Discovery of Quantification of Information Leaks", 30th IEEE Symposium on Security and Privacy, May, 2009.

over time randomness, or entropy as it is called, increases, and order decreases. But in the digital space, randomness processed through FSMs can at most retain its randomness, and when it does not, it decreases entropy and produces order. For physical space, it is "out of one, many" while for digital space, it is "out of many, one".

For the DFE examiner, claims with regard to the sequence of events that took place to produce incomplete traces are excessive unless they take into account these results, even if the precise FSMs involved are completely known to the examiner.

Many equivalent and similar FSMs

To here, the discussion has largely surrounded the properties of each individual FSM. But it is also important to understand that most modern computer systems are composites made up of many component FSMs, and that many of these components implement general purpose functions that can display a very wide range of behaviors. When composed, these FSMs produce a far larger set of composite behaviors. In essence, FSMs that are composed to form a general purpose computer are designed to model other FSMs, the specifications of which are “loaded” from inputs and may change with time. We loosely call these loaded FSM specifications “programs”, and may use “program” and “FSM” interchangeably.

An unlimited number of different FSMs may produce the same output sequence from the same or different input sequences. For example, at the level of computer programs in common use, an editor, electronic mailer, or user program, may produce the same outputs from different inputs. Thus:

$$\exists M_n:(I_n, O, S_n, I_n x S_n \rightarrow S_n', I_n x S_n \rightarrow O) \text{ for unlimited } n.$$

With incomplete traces, we cannot uniquely determine prior states and inputs. To the extent that traces are more or less complete, we may or may not be able to uniquely determine or bound the set of programs that might have produced the traces. We may not even be able to determine the extent of completeness of traces we have.

The number of input sequences per output and final state, (i.e., how many event sequences can produced a trace) can be calculated as $|I|/|OxS|$. For an n-bit input sequence producing a trace of m output

bits and s state bits, there is at least one trace producible by at least $2^{|O+S|-|I|}$ different input and state sequences:

$$|I|>(|O|+|S|) \rightarrow \exists(i,i')\in I:\exists(o)\in O,\exists(s)\in S,\ i\rightarrow(o,s) \text{ and } I'\rightarrow(o,s)$$

$$\text{and } |I|=2^n,\ |O|=2^m, n>m\rightarrow\exists i,i'\in I, o\in O\ i\rightarrow o,\ i'\rightarrow o$$

$$\text{and } \exists o\in O,\ \exists Ï\subset I,\ |Ï|\geq 2^{n-m}$$

This can be readily shown by assuming that all $o\in O$ stem from exactly $Ï\subset I$, $|Ï|=2^{n-m}$ traces. If we then remove $i\in Ï$ from any $o\in O$, that i must go into another Ï'. Thus it is guaranteed that some $o\in O$ must absorb that $i\in Ï$ and thus that $\exists o\in O,\ \exists Ï\subset I,\ |Ï|>2^{n-m}$.

The resulting traces are always bits

Another important side effect of the atomic nature of bits is that, regardless of any errors, faults, uncertainties, or underlying physical phenomena, the result of processing with an FSM is always in the form of bits. Regardless of the nature of the errors, when physical reality becomes the traces that form DFE, it is in the form bits. The nature of the physical mechanisms by which the bits came to be is no longer available by the time mechanisms produce the traces.

Resulting traces are always "exact"

Because results from FSMs are always in atomic units called bits, they are also always exact and precisely characterizable. For this reason, the values of the bits forming the traces should never be at issue in any DFE examination, and all parties should be able to come to a definitive agreement as to what the bits forming the traces are. While the interpretation of the bits may differ for one reason or another, the atomic nature of the bits, once the evidence is put into terms of bits, should be identical and agreed by all.

FSMs produce partially ordered output sequences

Output sequences from FSMs are always strictly ordered at some level of granularity, but the individual bits may be output at non-differentiable times, so that at the level of the bit, outputs are produced as partial orderings. A partial ordering is partial in that:

> (1) Either A occurs before B (A<B); A occurs after B (A>B); or A occurs at a time that cannot be determined to be before or after B (A≈B); and

(2) There is not always a first (INF: ∀A, INF<A) and there is not always a last (SUP: ∀A, SUP>A) item in time sequence.

Of course, A<B and B<A cannot both be true. A≈B does not mean that A<B or B<A is not true, only that which of these are true cannot be determined. When there are many FSMs involved, each FSM may produce a complete or partial ordering, but the combination of the results of the set of FSMs is a partial ordering.

Because the granularity of DFE is finite, no digital trace has infinite precision, and thus at below the level of precision of the traces, time cannot be differentiated or ordering determined.

Limits on accuracy and precision based on representation

Depending on how information is represented, representation may, by its nature, limit accuracy and precision. For example, most digital numerical processing is done in a representation with a fixed and finite number of bits, where each bit represents a power of 2. In decimal notion, we might have a number like 42.357, where the digits represent the tens, ones, tenths, hundredths, and thousandths places. In binary form, we might have a number like 10110.011 representing the 16s, 8s, 4s, 2s, 1, halves, quarters, and eighths places. If there are only a finite number of bits to the right of the decimal point, the number 1/3 cannot be represented accurately, either in decimal or in binary, in this format. If we then do mathematical operations on these representations, the results may extend those errors in the last bits or digits further and further, producing larger and larger errors, even though every computation was done perfectly. Similarly, a number that is too big to fit in a format may cause a result that cycles back to a lower number, becomes negative when adding two positive values, or behaves in other ways that are not intuitive in the physical world. There are other representations that are precise, but they are not always used. For example, the real number 1/3 can be represented as a structure as is sometimes done in LISP (e.g., "(/ 1 3)") or in other similar languages that process symbols rather than numbers. The precision can be extended to unlimited sizes as well, and numbers can be represented in arbitrary precision, within available space, at the cost of more time and space used in storage and processing.

The representations used will impact the accuracy and precision of results, and because digital space is discontinuous, at the margins, there may be complex errors that are rare and hard to predict. In addition, while there are ways to represent some real numbers with perfect precision, like certain transcendental numbers and numbers that can be represented by sequences, patterns, or programmed generating sets, there is no way to represent every real number (or even every integer) at unlimited precision in finite digital storage. The reason for this is that the quantity of unique integers ($|\mathfrak{I}|$) is $\aleph_0$ (there are an infinite number of them) and the quantity of unique real numbers ($|\mathfrak{R}|$) is $\aleph_1$ (an infinite number of them for each integer), while the size of the storage of any real computer is finite.

Information content in context and related issues

Shannon, in analyzing issues related to information theory, came up with a measure h(x) for the information content of a collection of symbols.[62] The notion underlying Shannon's approach is that information, by definition, is something that reduces uncertainty, also known as entropy. The content of a sequence of symbols in a language is therefore calculable as the degree to which it reduces entropy, or the difference between what is known by the sender and the receiver in terms of the relevant message. h(x) is calculated as:

$$h(x)=\log(1/p(x))$$

where p(x) is the probability of a symbol x occurring and the base of the log dictates the base of the result. For results in base 2 (i.e., yielding the number of bits of information), $\log_2$ is used.

Languages have different content density

To get a sense of this, for randomly chosen letters out of the English alphabet, ignoring capitalization, there are 26 letters, so h(x) is given by $\log_2(26)$, with the answer in base 2 of 4.7 bits. Of course symbols from the alphabet of the English and other human languages are not used at random, so for a more controlled syntax, the content differs for different letters. More common letters, such as 'e' and 'i' have less content because they do less to reduce uncertainty, while less common letters, like 'q' have more content

62 C. Shannon, A Mathematical Theory of Communications, Bell Systems Technical Journal. 3, no. 27, (July 1948).

because they are rarer. After a lot of statistical analysis of text, h(x) was empirically found to be 2.9 bits per character for English.[63] In other words, more than 1/3 of the symbols used to convey English are redundant. Th_s ex_mpl_ sh__ld mak_ th_ po_nt q__te cle_r.

It turns out that different human languages have different amounts of redundancy. For example, English, French, German, Japanese, and every other major language in use by people, each have differences in information content per bit, or as we call it, content density. Different species and different computer languages have different content density than spoken languages and than each other. These characteristics of languages may be used in forensics, among other things, to try to differentiate content types. As an example, file type detection rates of 92.1% with a false positive rate of 20.6% have been experimentally shown using this approach.[64]

But things are actually a bit more complex than this when trying to understand and characterize content. For example, computer languages tend to be different than human languages, because they are designed to be manipulated by FSMs, which are designed by people based on mathematical optimization methods and human concepts of what they prefer. It turns out, for example, that different codings, selections of symbol sets, and syntactic groupings, produce widely different results for content density.

Compression and other codings that alter content densities

Huffman, following Shannon's work, devised a method for constructing minimum redundancy codes.[65] His approach was to assign letters different bit sequences so that letters with more content had more bits, but appeared less often. This results in an optimal lossless compression of content[66] that is also self-

63 Ibid.

64 M. Karresand and N. Shahmehri, “File Type Identification of Data Fragments by Their Binary Structure”, Proceedings of the 2006 IEEE Workshop on Information Assurance, United States Military Academy, West Point, NY.

65 D. A. Huffman, "A Method for Construction of Minimum Redundancy Codes", Proc. I. R. E., 40, Sept, 1952.

66 D. A. Huffman, "Canonical Forms for Information-Lossless Finite-State Machines", IRE Trans. on Circuit Theory (special supplement) and IRE Trans. on Information Theory (special supplement) (1959), CT-6 and IT-5, pp41-59, May [A slightly revised version was in: E.F. Moore, Ed. "Sequential Machines: Selected Papers", Addison-Wesley, Reading, Massachusetts, 1964.

synchronizing,[67] and thus compensates for bit errors.[68,69] Similar methods are used for compression today, with some improvements. Codings may include such things as compression, but they may also include encryption, steganographic, and other “covert” forms.

Lossy and lossless transforms

When an FSM processes input and state to produce output and next state, the output may be related to the input in many different ways. One way to consider this operation is as a transform, or mapping, from input to output. Codings such as compression may be lossless, in that the output can be used to regenerate the unique original it was produced from, or lossy, in that the output cannot be used to regenerate the unique original it was produced from. This relates directly to the previous discussion of FSMs converging the digital space, in that a lossless transform does not converge the input space while a lossy one does. Neither retains the state sequence of the FSM, and thus they are both typically incomplete traces, but a lossless transform can be inverted to uniquely produce the previous input sequence, which can then, assuming a known initial state and FSM, be driven forward to produce an assumed complete trace. A lossy transform can only produce, at best, an envelope of prior input sequences, and from that envelope, again assuming a known initial state and FSM, a potentially very large set of candidate assumed complete traces.

Many transforms that are widely used are lossy. For example, when inputting from the physical world to the digital artifice, some maximum level of granularity is discernible based on the interface mechanism. All such transforms are lossy in that many possible physical situations within an envelope of possibilities may have led to the digital representation that results. Thus, for many possible inputs there is only one output. But that does not mean that any one input may have multiple outputs. The digital space, in general,

67 P. G. Neumann, "Error-Limiting Coding Using Information-Lossless Sequential Machines", IEEE Transactions on Information Theory, (Apr. 1964), vIT-10, pp108-115.

68 P. G. Neumann "Efficient Error-Limiting Variable-Length Codes", IRE Transactions on Information Theory, (Jul 1962), vIT-8, pp292-304.

69 P. G. Neumann, "On a Class of Efficient Error-Limiting Variable-Length Codes", IRE Transactions on Information Theory, (Sep 1962), vIT-8, pp260-266.

and with few well defined exceptions, is designed specifically to converge many inputs to one output. Thus the transforms of this sort are often many to one, while physical space is not this way.

Transforms that can produce any of the sequence of output bits of lengths based on input lengths, are called "onto" the output space, and transforms that produce outputs that prohibit all possible bit sequences from being produced are called "into" the output space. Thus digital transforms are generally either one-to-one or many-to-one and either onto or into. Many-to-one transforms are lossy, and uninvertible.

Hash functions and digital signatures as lossy examples

Hash functions and digital signatures are commonly used methods that are, in almost all cases, lossy. As a general rule, when the result of a transform is to reduce the number of bits regardless of the language characteristics of the input, the transform is lossy. This means that any given input bit sequence does not produce a unique digital signature or hash value. Rather, cryptographic checksums, hashes, digital signatures, and other similar methods that are often used to authenticate content, are not unique to the input. As shown earlier under the demonstration of “Digital space converges while physical space diverges”, given that the output is m bits long, there are only a total of 2^m possible values for that output. Given an input sequence of n bits, where n>m, there is at least one output value that has at least 2^{n-m} different input sequences that produce it (i.e., $\exists o \in O, \exists \ddot{I} \subset I, |\ddot{I}| \geq 2^{n-m}$). Since there are many different possible values of n for input sequences to a transform, the total number of inputs that could generate any given output is potentially enormous (i.e., $O(\sum 2^{n-m} \forall n)$).

Hash functions are, generally, designed to spread the input space evenly through the output space, so that the likelihood of a hash collision is equally small throughout the space. However, for some hashes and some input spaces, the input is not evenly distributed over the output, producing weaknesses. Cryptographic checksums are specifically intended to be designed so as to have this equal spreading property in order to assure their cryptographic utility.[70]

70 Scott Contini, Ron Steinfeld, Josef Pieprzyk, and Krystian Matusiewicz, “A Critical Look at Cryptographic Hash Function Literature“, Centre for Advanced

The utility of these transforms does not come from any uniqueness property. It comes from (1) the ratio of number of input sequences not producing a given output sequence to the number of inputs producing that same output sequence, (2) the evenness with which the transform spreads nearly identical input sequences over the output space, and (3) the computational complexity of intentionally creating an input sequence that will produce a given output sequence, possibly also in a valid syntax and with meaningful content. Computational complexity will be discussed later.

Content only has meaning in context

A sequence of bits may have multiple meanings in different interpretation schemes. The same sequence may be interpreted with different FSMs in different environments, have different meanings to different people, and produce different state sequences and traces in different FSMs. This introduces the notion that bit sequences only have meaning in context.

One worthwhile result is that, for any particular language, you cannot get more than one bit of content per bit of message. We will see the impact of this shortly, and other effects will also be apparent later. But just because there can be no more than one bit of content (fundamentally a single definitive Boolean decision) per bit of state, does not mean that different interpretations of the same bit cannot have different results in different environments.

Semantic information content

There is no uniform theory of meaning for information, and to the extent that such a theory is ever created, it is, for the moment, beyond the scope of digital forensics, except in one way. There is an underlying meaning of bits in that they are interpreted by FSMs, and those FSMs produce state changes and outputs. The context for digital content is the FSMs that process that content. Their meaning in the context of those FSMs may be interpreted as the resulting state and output sequences. Thus, the meaning of a computer program may be considered in terms of its execution histories in the environment in which it runs, or in other words, the set of traces it produces.

Computing, Algorithms and Cryptography, Department of Computing, Macquarie University. ECRYPT Hash Workshop, May 2007.

Eats shoots and leaves

The interpretation of symbol sequences can be uncertain. But since FSMs always move from a current situation forward in a systematic way, they will always come to a defined future state and set of outputs regardless of clarity in inputs from a semantic point of view. While FSMs can be designed to produce states and outputs that include "don't know" sequences, each such “don't know” state is realized as a particular setting of bits in implementation. As a result, the FSMs still execute forward in a predictable way, even if the specifics depend on the specific design and/or implementation rather than the specification.

FSMs are limited by the ability of people to design and implement them. If the designers didn't anticipate or otherwise provide for various input sequences or states, the FSMs will continue to operate nonetheless, producing outputs that are predictable, but may not have been predicted by those responsible for creating them. In some cases, such FSMs enter unanticipated states that they cannot leave, enter states that are closed subsets of the total set of states, or enter states that produce unanticipated and undesired output sequences from desired input sequences.

Returning to phrases like “eats, shoots, and leaves”, people tend to interpret such phrases in light of their "point of view", but FSMs don't have "points of view", they have states. Regardless of claims put forth by those claiming to have intelligent computers, to date, they have not met or approached the human ability to deal with uncertainties of this sort. Many people anthropomorphize when they make statements about computers, but this is inappropriate for a digital forensics examiner in the context of an examination. It may take a few extra words to describe mechanisms in terms of the operations of FSMs processing input sequences and states to yield next state and output sequences, but it is accurate and appropriate.

How computers work and their limits

Computers we normally see in digital forensics are complex composites of components, largely consisting of physical mechanisms that realize FSMs, input and output (I/O) devices, and supporting mechanisms. From a standpoint of DFE examination, we will largely ignore the physicality of the mechanisms and focus

on the limits of the composition of FSMs and limitations associated with the fact that FSMs are implemented in physical mechanisms.

The DFE examiner needs to know how computers work in order to be able to understand how traces come to be and to make sound judgments about issues in the examination. At the same time, the examiner that believes that they know everything about how computers work, is more likely to be a liability than an asset, because, as a science, DFE examination requires that the examiner keep an open mind and not overstate things.

There is simply too much to know about computers for any one person to know it all. As such, the examiner is in a constant battle with technology to keep up to date in specialty areas where they work, and to learn enough about other areas to be able to learn more. They must also do experimental work to make more definitive determinations in cases where their personal expertise is not as definitive as the situation demands.

From hardware to FSMs

At the hardware level, computers are composed of electronic or other circuits that act based on the physics of devices to process signals in one form or another. These hardware mechanisms are ultimately designed and implemented so as to represent and operate on binary values (bits). The hardware that processes bits is often referred to as an automaton (plural automata). Mechanisms that process sets of input bits into output bits without storage are called combinational logic circuits. Mechanisms that take input bits, combine them with stored bits, and produce output while updating the stored bits according to a predefined method are called finite state automata or machines (FSMs).[71,72] In FSMs, we may reasonably think of the predefined method as the "program" and the inputs and outputs as "data". The program interprets input data to update the stored state and produce output data.

71 E. F. Moore, “Gedanken experiments on sequential machines,” Automata Studies. Princeton, N. J.: Princeton University Press, 1956, pp. 129-153.

72 G. Mealy, "A Method for Synthesizing Sequential Circuits". Bell Systems Technical Journal 34: 1045–1079, 1955

Program or data - what's the difference?

Computation in the sense most computer scientists see computers is understood in terms of a paper by Alan Turing[73] in which, among other things, he defined a notional computing machine consisting of an FSM and an infinite length “tape”. It operates by reading a symbol ($i \in I$) from the tape, updating its state, writing its output ($o \in O$, $O \approx I$) to the tape, and moving the tape left or right. Turing identified an FSM for such a machine that could compute any function that any other such machine could compute, called a Universal Computing Machine, a Universal Turing Machine (UTM), or simply a Turing machine. This theoretical model is very useful for addressing limits and understanding other issues.

Turing's approach was to use a single predefined method to describe any other FSM, with the details of the particular modeled FSM described by the stored data states at the time the UTM starts. Thus, except for limits on performance, any transformation of input sequence to output sequence that could be implemented in any FSM could be modeled accurately with any UTM. It turns out that, since the UTM itself is just a predefined method with an unlimited number of memory states, a UTM can contain a "program" that is, itself, a UTM (thus “universal”). The term "Turing capability" or “general purpose” is often used to describe this nature of computer systems. Notionally, any general purpose computer can model any other general purpose computer, and this notion goes on recursively. This is why "virtual machines" are possible, in which an Apple computer running the OS/X operating system can can run an emulation of a PC running the Windows operating system, etc.

Nearly perfect virtualization and simulation are possible.

The notion that stored states are descriptions of FSMs is identified with the term "stored program computer" in that the memory states store the "program" that describes the FSM that the UTM models. Stored programs are typically characterized by sets of "instructions" that take "input", act on "memory", and produce "output". The

73 A. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem", London Math Soc. Ser 2. Vol 42,Nov 12,1936, 230-265.

"instruction set" of the computer is the set of stored values that are interpreted by the mechanism to perform various "instructions".[74] One computer's "data" is another computer's "instruction".

General and special purpose computers

Not all FSMs have Turing capability. We may differentiate between "general purpose" and "special purpose" computers in that general purpose computers have (finite) Turing capability, while special purpose computers do not:

> A **general purpose** computer **CAN be programmed** so as **to perform any function** a finite UTM can perform.
>
> A **special purpose** computer **CANNOT be programmed** so as **to perform any function** a finite UTM can perform.

General and special purpose computers

Many, but not all, computers, are general purpose. General purpose devices are programmable in the sense that stored values can be changed, either by replacement of a hardware device or by setting of the stored values through software, and this allows them to be reprogrammed for other purposes. For general purpose devices, the examiner has to understand both the workings of the device and the workings of the stored states. To the extent that the stored states are interpreted in a general purpose way and themselves implement a recursive interpretation mechanism, understanding the operation of the overall mechanism may require, or at least be facilitate by, understanding all of the recursively implemented machines, including their interactions.

Special purpose digital devices also exist, and in large numbers. For example, many digital watches, input and output devices, and control mechanisms are special purpose in that they can never be programmed or altered so as to be able to perform general purpose computational functions. DFE can come from special purpose devices. In order to examine such traces, the examiner must know how the device works and where the traces were stored within the device. Based on that knowledge, the examiner may interpret the

74 A. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem", London Math Soc. Ser 2. Vol 42,Nov 12,1936, 230-265.

meaning of traces in terms of what the device does under what circumstances.

To the extent that general purpose computers include states that produce special purpose functions, the special purpose functions embedded within the general purpose functions must be understood as well in order to understand the overall function.

Special and general purpose operating environments
Many, but not all, general purpose computers use general purpose operating systems and related content to control their execution. Typical examples of special purpose implementations are micro-controllers, such as the programmable logic arrays used for high-speed encryption and decryption, visual image processors, etc.

One of the most important examples of a special purpose operating environment is the sort of programmable logic controller used to limit the operation of devices in some infrastructure systems, mechanisms that control movements of physical devices, doses of radiation, and so forth. These devices are designed to allow complex controls over complex machinery, but are specifically limited in their programming and programmability so that they do not allow the physical devices to go outside of specific controlled boundaries, to move too quickly or too slowly, to hit themselves, or to create hazardous conditions for the rest of their environment. While such systems are increasingly being replaced by general purpose operating environments, they continue to be used and provide higher surety in exchange for less programmability.

Many devices, such a telephone switches, cellular telephones, mobile phones with base stations, copiers, printers, telephone answering machines, and so forth, have either special purpose processors or general purpose processors with special purpose operating environments. These operating environments and their associated special hardware, work together according to their implementation. The examiner has to know specific details of how they operate in order to meaningfully interpret the sequences of bits that form the traces associated with them, in the same way as special purpose devices require such knowledge. The same is true for many satellite control systems, systems within automobiles, aeronautical systems, and other similar embedded systems.

Some of these devices, and almost all personal computers, mainframe computers, minicomputers, and many of the other digital devices of the sorts in use today, use general purpose operating systems. They implement their functions through programs that execute from within the operating environment provided by those operating systems, system libraries, and the hardware devices in which the operating system functions.

Special and general purpose programs and interfaces

Many, but not all, programs in widespread use, provide special purpose function at most or all of their interfaces. Typical examples of special purpose programs include Web servers, calendar applications, music and video recording and playback mechanisms, and most applications that users interact with on appliances, like telephones and ticketing systems. Typical examples of special purpose interfaces include menu systems and work flow systems, which provide interfaces that access underlying general purpose mechanisms but limit the interactions with those systems to limit user functionality.

Most programs that provide general purpose interfaces are called "programmable" and typically have defined languages by which they are programmed. For example, LISP, Perl, C, Basic, and other similar languages are designed and intended to be programmed, and provide both general purpose functionality and the ability to implement special purpose interfaces. Other programs, like many modern spreadsheet programs and document processors include programmable functions, often called “macros” or with similar names.

Because of the desire to provide many more services via Web interfaces, many Web browsers today include “plug-in” capabilities for programming languages, such as Java and Javascript, that allow remote content to be loaded as data and run as a program. Other plug-in capabilities, such as programs to display special formats (e.g., PDF viewers, video viewers, etc.) may be intended to provide only limited function, but have programming flaws that cause them to operate as general purpose programs when certain input sequences appear.

DFE examiners who encounter such programs may end up with a very broad scope of examination in cases where programs or interfaces intended to be special purpose are exploited so as to use general purpose functionality.

As in the case of virtual systems, interactions between special and general purpose programs and interfaces can greatly complicate understanding, producing large numbers of very diverse input and state sequences associated with produced traces.

Processes, files, and other structures in computers

General purpose operating systems commonly provide structures that abstract the hardware to a large extent so as to allow the programmer to write general purpose applications that can execute in different hardware, be ported from environment to environment over time, operate in different hardware platforms, and be used in conjunction with other hardware and software to build up larger and more complex systems that perform multiple functions.

In order to do this, the operating environments use certain common abstractions, typically including, at a minimum, processes, files, input and output streams, separation, and control mechanisms. DFE examiners who work with traces from these environments must understand both the broad notions of processes, files, and other operating system structures, and be able to get at the details of the mechanisms in particular environments in order to be able to interpret some of the traces properly.

For example, at the level of files, some operating systems support transactional integrity so that appended contents from multiple processes do not get intermixed. In such a system, when two activities take place nearly simultaneously, the one that first gets the lock on a common append-only file, will have exclusive access until it is done writing, and then the next process will get control. As a result, the examiner that sees intermixing of results from different processes can determine that this is inconsistent with the normal functioning of those processes in that type of file access in that operating system. Similarly, a process awaiting a locked file may have pending outputs containing time stamps that end up added after subsequent timestamps are generated and added by another process, creating an apparent, but false, time sequence error.

Very few DFE examiners will be experts at understanding the detailed issues of more than a few operating environments, and clearly this level of expertise is required in order to examine and properly interpret traces in these environments at this level of detail. Of course reconstruction is an approach to resolving such issues, but only if the examiner knows enough to know that a reconstruction is called for, and what to try to reconstruct to resolve the issues at hand in the case.

Higher level structures

Within operating systems there may be other structures. To the extent that these structures are similar to operating systems in that they abstract underlying structures, they too impact the examination process and bring additional information to the examiner seeking to analyze and interpret traces resulting from those structures.

For example, the "Java Virtual Machine" (JVM) environment, the "LISP" interpreted environment, the "Basic" interpreted language, and each of the different versions of these and other similar languages, provide additional structure that effects traces that are produced, and limits what can take place consistent with the use of that environment. These added structures provide additional information to the knowledgeable examiner who can then understand consistencies and inconsistencies in these contexts as well. And in each of these language environments, programs create additional environments that further structure and limit the traces that are consistent with their execution.

Just as the recursive nature of Turing capability applies to all FSMs and programs, the issues associated with the nature of and interactions between states under control of those FSMs and programs and traces produced and retained by them, may all impact the examination process and the manner in which traces are properly understood.

Similarly, each of these layers may introduce new concepts that have effects on the “physics” of information within the context of those environments. These layers will not change the underlying nature of the physics of digital information any more than the physics of digital information changes the underlying nature of natural world physics. But the ability to analyze and understand the

issues of the legal matter in the context of the additional layers may lead to additional understanding and insights regarding these environments that are not present in the underlying physics of digital information.

As a simple example, many transaction engines produce complete traces of all input sequences to the transaction engine through an audit mechanism, and provide for replay including sequencing. This means that, at the transaction level, complete traces are present.

The nature and challenges of composition

As described above, most digital computers today are composites made up of component FSMs interconnected through interfaces. As exemplified by the select details on how computers work, the low-level physics of digital information interacts significantly with the higher level concepts and structures, and multiple FSMs executing in parallel, produce potential interactions that may substantially effect the use of the underlying physics.

Composition is not a well understood issue in the computing field, and to a large extent, the success of composition at the hardware level stems from strict controls over sequencing of and interactions between FSMs by computer engineers designing digital systems. As an example, a simple central processor unit (CPU) will typically have a single FSM that controls the other FSMs so that they only interact with each other under conditions set by the controller. A hardware failure that permits other interactions may range in consequences from producing unreliable operation of the processor (e.g., it fails to perform some of its state changes properly) to an electrical event that destroys the processor and produces smoke, sounds, and smells for the human observer.

While the CPU designer has more or less complete control over how components interact to form the composite, in a general purpose computer, the set of computer programs operating at one time is controlled largely by the user's usage patterns, and the designers of the individual hardware and software components often have no idea of what other hardware and software will be present. At the hardware level, specifications typically identify the constraints required to allow safe and properly controlled interaction, but in software, few if any such specifications exist, and

control mechanisms are far too complex to even be completely tested over the range of expected operating conditions.

For these and related reasons, compositions that are commonly present in widely used operating environments result in substantial effects that can be seen and understood in terms of the notions of the physics of digital information, and these effects can and should be understood by the DFE examiner when they are relevant to the issues at hand in a legal matter.

As an example of the effects of time interactions in composites, the generation of timestamps and other traces generated by processes is a natural starting place. As described above, time in digital systems forms a partially ordered set (POset). In order to properly understand the meaning of timestamps, we need to understand the accuracy and precision of those timestamps, and this ultimately drives toward understanding granularity. The term Δ was used to describe the minimum granularity of a timestamp, and in order to identify ordering in time, we must know its value for the particular environment producing the traces at issue. The problem is that, in an environment in which timestamps are produced by processes that interact, potentially arbitrary delays are possible between a process generating a timestamp and writing of that timestamp into a storage location that is part of the incomplete trace available to the examiner. Experimental methods might reasonably be tried to try to determine reasonable values for Δ, and such experiments have been done from time to time, with results in some environments ranging into days, depending on the particulars.

Another example is the challenge of trying to limit the envelope of inputs associated with traces. While, in general, digital space converges with time, the extent size of the input space producing particular traces may be reasonably limited in well controlled environments. But as interactions between components enters into the picture, event sequences producing any given trace expand rapidly. The question that remains is whether the differences between different event sequences is material to the matter at hand. For example, if there were an enormous number of different event sequences producing the identical trace, but all of those event sequences were consistent with a theory of the case and inconsistent with other identified theories, the forward convergence

would not be a substantial challenge. But when there are large numbers of FSMs present and sequencing of events at the level of those FSMs is an issue, compositions become a more serious challenge. For example, an unpredictable and untraced external event may trigger completely different internal sequences when events close in time happen in a different sequence, priority interrupts in hardware may drop one input and retain another, and incomplete traces may lead to unknown orderings with a high degree of consequence on envelope of alternative histories.

Computational complexity: a different "speed of light"

There are limits on what FSMs and digital mechanisms can do, and these drive the analytical frameworks used in the examination of DFE. In particular, while the speed of light has a physical effect on the movement of bits from place to place, there is a different sort of equivalence to the speed of light for digital transformation by FSMs. That equivalence deals with time and space, and it is called computational complexity.

Computational complexity is a theoretical construct backed up by mathematical results that show that, under currently understood mechanisms, the transformation of one thing into another via digital methods requires time and space proportional to the mathematical properties of the transform. It further asserts that certain things take certain amounts of space-time and that time and space can be traded off for each other. For example, something that takes 100 time steps of an FSM with one bit of input and state might be accomplished in one time step with an FSM containing 1000 input and state bits, in the proper configuration.

Two particularly useful aspects of computational complexity in DFE examination are discussed here. One aspect is that computational complexity limits what can and cannot be done in any particular FSM, and thus limits the causes of particular effects. The other aspect is that computational complexity limits what the examiner can do to examine a given collection of traces with a given set of resources.

Limits of what can be done (decidability)

The mathematics of computational complexity substantially moved forward with a paper by Alan Turing that, in addition to defining the UTM as a computational model, described a class of computational problems that could not be solved in finite time by any UTM (or by extension FSM), even if it had an unlimited amount of storage.[75]

The "halting problem" is the problem of having one computer that can decide whether another computer running any particular program will ever stop. It turns out that if you cannot solve this problem, there are lots of other problems that cannot be solved, such as writing a program to accurately detect all and only computer viruses in finite time.[76] It also turns out that, just as there are mathematical problems that can never be solved because no decision procedure can ever be found,[77] there is no such decision procedure for UTMs.[78] Therefore, this is also true of perfect computer virus detection, perfect intrusion detection, and many other similar problems, including many problems that may face DFE examiners. These problems are called "undecidable" because no decision procedure exists or can ever exist to always correctly solve them in finite time.

Certain problems are undecidable.

Decidability only applies when there is an infinite storage capability. For all realized digital systems, there is only finite total available storage, and thus these results are not strictly true for FSMs. But in practice, the difficulty of solving these problems may be so high that no solution is ever likely to be found for a large portion of cases.

It also turn out that, for the general class of digital systems with unlimited storage, a system cannot be both consistent and complete.[79] Consistency implies that all propositions always yield

75 A. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem", London Math Soc. Ser 2. Vol 42,Nov 12,1936, 230-265.
76 F. Cohen, "Computer Viruses", ASP Press, 1985.
77 S. C. Kleene, Meta Mathematics, 1952, North Holland Press pp.136-137.
78 A. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem", London Math Soc. Ser 2. Vol 42,Nov 12,1936, 230-265.
79 S. C. Kleene, Meta Mathematics, 1952, North Holland Press pp.136-137.

the same results regardless of how they are addressed, and completeness implies that all propositions that are true can be determined to be true by a finite number of steps. Just as the issues of decidability apply to systems with unlimited storage, real systems have limited storage, and therefore these mechanisms can be, and usually are, both consistent and complete. But this does not make it feasible to solve all problems in the digital world.

Computational complexity

In addition to the undecidable problems, there are also problems that are harder or easier to solve than other problems. For example, the famous "traveling salesman" problem has, as its objective, to find the shortest distance traveling salesmen can travel while visiting each of their customers. This is similar to a wide range of other problems in optimization. It turns out that, despite long-term research by many mathematicians and other experts, no way has ever been found to always either solve this problem, or test if a given route is the most efficient, in "polynomial time" in the number of places the salesman has to visit. Polynomial time means that, the time can be expressed as a polynomial, like $ax^i+bx^j+cx^k+...$ where x is the number of places the salesman must visit. In mathematics, this is expressed as "NP-complete", which stands for "nondeterministically polynomial time complete". Of course there are special cases, such as all of the customers being arranged around a single block, but in general there is no polynomial time solution known.

Generally, the notation used to discuss computational complexity is O(x) where O stands for "order of magnitude" and "x" is replaced by an expression. For example, O(1) is something that can be done in a fixed time regardless of the number of things being addressed. O(n) means that some constant number of steps per item is required, so it takes the number of steps times the number of items to get the answer. Generally, for large enough problems, where the number of items considered is substantial, we know that $O(1) < O(\log(n)) < O(n) < O(n \cdot \log(n)) < O(n^2) < O(x^n) < NP < O(n!) < O(n^n)$, etc.[80]

80 Mathematicians have identified that as $n \rightarrow \infty$, $x^n \rightarrow n!$, but as a practical matter, the complexity for finite computations can be considered different and n! always gets larger faster than x^n as n increases beyond 2x for positive x,n.

As the complexity of the problem gets higher, it becomes less and less practical to solve for substantial input sizes. Generally, when the complexity gets to be higher than polynomial time, it is impractical to solve it for problem sizes that are commonly encountered. For example, suppose it takes a nano-second (10^{-9} seconds) for each step, and a problem is exponential (it takes 10^n steps for n items). If n is 1, it's very fast, but if n is 10, then it takes 1 second, and if n is 100, it takes 10^{10} seconds, or 10 billion seconds, which is about 300 years. If n is 1000, even with all of the computers in the world working for thousands of years, the solution will be nowhere close.

This notion of computational complexity is important to the examiner in that it provides insight into things that are worth doing and not worth doing as well as insight into things that could have been done and things that could not have been done to produce the DFE. And that is very helpful in both deciding what to do and what was done.

Limits on the examiner from computational complexity

Limits on the examiner are generally imposed by the computational complexity of the sorts of things they want to do in their examinations. It is often helpful to do a quick calculation, or a quick test on a small sample, before proceeding to a full scale execution of a forensic examination technique associated with DFE, because full scale analysis often takes a lot of time. For example, if the examiner wants to search a collection of evidence for some specific sequence of characters, this will take only linear time, which is O(n) steps. But if the examiner wants to search for all possible computer viruses or Trojan horses that could have been put into the computer that produced the evidence, the problem is undecidable, and even starting the search is a waste of time. The examiner might try a different strategy, like searching for a known collection of Trojan horse programs. But this also means that the examiner has to understand the limits of the search methodology in use and that the results will not be definitive as to the absence of Trojan horse programs. Rather, it will only indicate that none of the known Trojan horses associated with the technique used were found.

At the same time, the examiner who understands these issues can better evaluate the statements and examination results of other examiners in light of this knowledge. For example, if an examiner indicates that they have done something that seems too complex to actually be done with the available resources, or something that is, in general, undecidable, then the claims should be very closely examined and refuted unless they are in fact shown to be true. The laws of information physics in general, and in particular, the mathematical results on computational complexity, apply to everyone. The discussion of computational complexity will continue later in more depth in terms of how long it takes to do what sorts of examination processes, but for now, it is enough to know that the examiners on all sides are limited by it.

Limits on the evidence and statements about it

In addition to limits on examiners, there are limits on everyone and everything else. These limits are particularly interesting when applied to evidence and statements made about evidence.

Suppose party A proffers a graphical image file (G) purported to contain hidden information (H) that party B is accused of illicitly leaking. The DFE examiner should be able to either confirm or refute this contention based on the available traces and the basis for the claim given by party A.

Refuting such a claim involves showing that G does not contain any bit sequences that represent H. But any sequence could represent H, depending on how that sequence is interpreted. That's why the party asserting a claim typically has the responsibility to show why their claim is true. Suppose that the total information content of H, even after optimal compression, is an order of magnitude larger than the total available bits in G. Then, except for a special case in which the interpretation mechanism already has large subsets of H identified, G cannot contain H. In this case, the claim may be largely refuted almost regardless of how it is shown. But suppose A's claim shows that B has an FSM with a coding such that H could fit into G. As the set of issues go back and forth, different differentiating factors may be found based on the specifics of the claims and their wording, and the specifics of the DFE.

The key thing to get from this discussion is the general concept that using these theoretical notions, a lot of information about evidence, as it applies to any particular case, may be gleaned. But it can only be gleaned with a clear understanding of the underlying information physics.

How many FSMs produce identical or nearly identical results

Many FSMs, given an identical initial state and input sequence, may produce an identical final state and output sequence, many more may produce almost identical outputs, many more may produce identical outputs with slightly different initial and final states, and the expansion continues at an increasing rate as the similarity criteria expand.[81,82] The exact number of different similar FSMs for a typical computer is not known at this time, but it is certainly at least $O(x^n)$ where X is a constant and n is the number of instructions.

For some known equivalence classes, the number of equivalent machines is O(n!) where n is the number of "statements", as an example, in a series of linear independent operational steps, where the ordering of steps may be arbitrarily changed.[83]

There are an unlimited number of FSMs that produce equivalent outputs for some subset of their equivalent inputs but that have many different input sequences for which they act very differently. Arbitrarily many can be generated by taking an identical FSM and adding as many states as desired, each with unique state change conditions and outputs.

Given this large (unlimited) number of equivalent and nearly equivalent FSMs, the examiner may not be able to uniquely identify which of that set of FSMs produced a particular trace within a computer. While the presence of the coding for a particular FSM in a trace may seem like a compelling reason to identify an effect with that FSM as the cause, there may be, and typically are, many other

81 F. Cohen, "Operating Systems Protection Through Program Evolution", IFIP-TC11 Computers and Security (1993) V12#6 (Oct. 1993) pp.565 – 584.

82 A. Friedman and P. Menon, "Theory and Design of Switching Circuits", Compuer Science Press, Woodland Hills, CA, 1975.

83 F. Cohen, "Operating Systems Protection Through Program Evolution", IFIP-TC11 Computers and Security (1993) V12#6 (Oct. 1993) pp.565 - 584

interpretation mechanisms in operation at the same time, and there are almost certainly many different input sequences and FSMs that can produce equivalent results.

This same result is also helpful to the examiner in that it means that there are many equivalent approaches to getting the same analytical result from the DFE examiner's tools. Thus any potential flaws in one tool may be detected by the use of other tools that are equivalent in their results for some set of analytical processes.

Designs that take advantage of complexity

The notions underlying computational complexity are the basis for using computational leverage to advantage. For example, using current known methods, it is generally far easier to create and multiply large prime numbers than to factor the product of large prime numbers. This is used in cryptography to generate codes that are very hard to “break” without the key, but relatively easy to use. The basis for this class of systems is that it's easier to make than to break, and if that computational differential is high enough, then what can be practically made may not be practically “broken”.

Public key cryptography, cryptographic hash functions, and other similar methods are based on this notion, which has its roots in computational complexity. But caution must be used in claiming things about such systems from a standpoint of DFE examination. Just because something is hard does not make it impossible, the computational complexity arguments are uniformly based on assumptions that are not uniformly true, and they all assume that all of the action takes place within the artifice of the digital system, which it may not.

As a simple example, a trace of a document and its valid digital signature using a public key does not mean that the named owner of the associated private key signed the document. At most it indicates that the identified private key was used to sign it.

Outside the artifice

It is vital to understand that all of the information physics discussed here assumes that the digital systems involved are contained within the artifice - the digital space. But in reality, the digital space is implemented imperfectly in the physical space, which is not subject

to the constraints of the artifice. Any or all of the assumptions about the digital space may be violated in the physical space.

Fault tolerant computing and testing

The field of fault tolerant computing and the closely related field of digital system testing focus in on methodologies for detecting and mitigating different sorts of faults in digital systems, including a historical focus on those faults that occur in the physical realization of digital space, within the field often called computer engineering. A large body of literature exists on the types of faults that are commonly observed and their causes.[84,85,86] The design of digital systems is commonly done so as to provide defined fault rates in terms of measures like the mean time to failure (MTTF) from those sorts of faults. Tests are designed to be able to detect specific types of faults to defined levels of coverage so that, for example, 100% coverage of stuck-at faults can be achieved in many digital devices. The reason digital systems are very reliable and rarely make hardware mistakes is that the computer engineering field has spent a great deal of time and money in (1) seeking out and mitigating these sorts of faults, (2) improving processes so as to reduce these types of faults, (3) testing products to find these sorts of faults prior to deployment, and (4) making fault tolerant systems that can compensate for some number of some types of faults before producing a failure in the sense of a wrong output from the FSM that produces the final output of the device.

Accidental violations of digital space assumptions

Despite all of the efforts to design digital systems in a reliable way, there are still many accidental causes of faults at the hardware level, such as high levels of external electromagnetic interference or temperatures out of normal operating range, that can cause hardware failures. But even when the digital devices are working

84 M. Breuer, A. Friedman, "Diagnosis and Reliable Design of Digital Systems", M. A. Breuer and A. D. Friedman, Computer Science Press, 1981, Breuer, Rockville, Md.

85 Melvin A. Breuer, "General Survey of Design Automation of Digital Computers", #1710 Proceeding of the IEEE, December, 1966.

86 The International Test Conference and many other conferences and venues consistently examine built-in self-test and a wide range of related methods. http://www.itctestweek.org/history.shtml

properly, there are other accidental causes of violations of assumptions made in different sorts of systems and approaches.

For example, the assumption that cryptographic keys are generated at random, commonly required for a system with high complexity leverage, is almost always wrong in practice. This may result in mechanisms that do not have the computational leverage they are assumed to have.[87,88] Software is often designed without perfectly matching its specification, specifications often fail to realize the concepts the originator had in mind, and interferences like timing problems or inadequate storage space commonly cause the mechanisms to fail at times and in ways not anticipated by their designers, implementers, or users. When these sorts of things happen, assumptions are no longer true, and the physics of digital space may no longer be fully in effect. The DFE examiner must recognize that these kinds of things happen and take appropriate precautions in their examination and in the reporting of results.

Intentional violations of digital space assumptions

Malicious actors of all sorts violate the standard assumptions of the digital space and/or the typical assumptions about how systems operate on an ongoing basis. The vast majority of attacks that are publicized are "computer security" attacks in which someone exploits a software or usage error in a digital system, thus causing the FSMs to act differently than originally intended. These sorts of intentional acts all follow the laws of the digital space, but they may violate many of the common assumptions regarding the uses of the space. For example, computational complexity arguments about the difficulty of forging, creating, or bypassing some mechanism are no longer valid if the malicious actor has access to the same mechanisms as the originators of the content. They may alter the content being subjected to those mechanisms, use private keys to sign on behalf of the real user, alter the presentation of the results of forensic analysis, or any number of other things. The potential for

87 Kerberos is an example of a system that had a pseudo-random number generator that was predictable as a function of time, resulting in an attack against the overall system.

88 See also "Security of Random Number Generation: An Annotated Bibliography" at http://www.cs.virginia.edu/~rjg7v/annotated.html and http://cobweb.ecn.purdue.edu/~kak/compsec/NewLectures/Lecture10.pdf

these sorts of violations must be recognized by the DFE examiner in making statements about the accuracy of their results. To get a sense of the magnitude of this situation, there are single actors who have released computer viruses that have altered the execution of FSMs in more than 10 million computers and sustained these and further alterations over periods of years without being caught.[89]

The less common intentional violation is the direct violation of digital space assumptions by malicious actors. For example, the hardware devices that implement digital space may be modified or contain Trojan horses that cause them to violate the rules of digital space, the assumption that all information passes through digital systems may be violated by the use of elicitation techniques against humans, covert channels not in the digital space may be exploited, and signals may be induced into the digital systems to cause them to fail in attacker-desired ways. Again, the assumptions of digital space must be considered by the DFE examiner, and any report of results should take into account the potential for intentional alteration of the digital space via non-digital means.

Where worlds collide - the interface

Because the digital world is discontinuous and the physical world is continuous, the digital world tends to both amplify infinitesimal differences near discontinuities and suppress more substantial differences far from discontinuities.

At the interface between digital space and physical space, there are any number of opportunities for error. At the simplest level, the reader may be unable to differentiate the 2 from the 2 in this book, but rest assured, these two instances of the depiction of the symbol we commonly use for showing the number two are quite different. They are in different but similar fonts. In fact, one of the fonts may not be available on any given digital system while the other may be available on it. This simple instance of an interface interpretation error demonstrates that, at the interface between digital space and physical space, any number of faults may be produced that may lead to erroneous interpretations. All of the cognitive faults of

89 Phillip Porras and Hassen Sa¨ıdi and Vinod Yegneswaran, "A Multi-perspective Analysis of the Storm (Peacomm) Worm", SRI International, CSL Technical Note October 10, 2007.

people are potentially exploitable at the interface, but this is only the beginning.

Incoming physical signals lead to digital signals through translation devices. These physical to digital devices must, by the nature of the differences between the worlds, introduce error, in that they can never capture the precise physical information at the infinite granularity of the physical world. Tracking the errors in this interface through to the digital system is not generally done, so FSMs at the interface ignore errors and accumulate them as processing is done.

Physical values near the bounds of digital differentiation may also produce time delays at the digital interface. Nearly identical physical values may be translated into the digital world as substantially different while greater physical differences may be translated as identical into the digital world.

One of the most common challenges faced by the DFE examiner is the fact that the appearance of a picture may be so similar to another picture that they seem identical. But when comparing images at the bit level, there are often substantial differences. In general, notions of "similarity" of this sort are problematic. Even two images with identical bit representations may be generated with different methods or captured from different locations. Two images created at almost the same time and place and depicting the same scene may generate different digital representations, and even the same physical world image may produce different digital values when the identical methods are applied under well controlled circumstances.[90] The same is true, in general, of any physical state translated into digital state.

These limits also apply to issues of time across the digital and physical worlds. Digital world time is granular to the granularity of the FSM mechanisms, while physical world time is, according to the physics of today, continuous. In the digital world, everything that

90 As an experiment, we repeatedly scanned the same piece of paper 9 times on the same flatbed scanner without delay and without moving the page. Each resulting file varied in length and content from every other scan. At the level of 16 byte chunks, the files differed in 99.96% of chunks. The first 256 bytes were identical headers, and only 153 other chunks matched across files, these matches were distributed throughout the files and across different pairs of files.

occurs takes finite and bounded time, with travel limited by interface delays and distances, and time granularity limited by clock rates of the FSMs. Physical events that take place too quickly for the digital space may even be completely missed.

The speed of light limits physical world information exchanges except possibly for quantum entangled physical objects, and physics is still unsettled on that issues. The speed of light limits in physical space also limit FSMs both internally and when they communicate with each other.

Near simultaneous physical events may appear to be simultaneous or in different order in the digital world and simultaneous events in the physical world may appear to be non-simultaneous in the digital world. Thus, at fine levels of granularity, even the ordering of physical events in time may be misrepresented through the analog to digital interface.

What sensors sense and actuators actuate

Another problem at the interface between the digital and physical worlds is that sensors that interface between the worlds deal only in a limited set of parameters.

For example, there are many properties of a physical surface that may be detected by various methods and under various conditions. Different lighting may reveal hidden characters on a printed document, and holographic images may appear differently depending on the angle from which they are viewed and the orientation of external lighting. A surface may be rough or smooth, sticky or slippery, hot or cold, wet or dry, and so forth. But the digital sensors that read the physical items only translate what they sense into the digital world. If the sensor cannot tell how wet, dry, hot, cold, sticky, slippery, rough, or smooth something is, it will not be reflected in digital space.

In the actuator realm, the same limitations exist, but in a different way. If the output does not control temperature or dampness, slickness or stickiness, or other physical parameters, the digital output may produce arbitrary values within a range of the capabilities of the physical output device and in the context of the external environment, and these will not reflect the digital output. If

some of these are controlled and others are not, then the controlled parameters may effect the uncontrolled parameters, and the digital system will be unable to control those interactions.

The combination of output and input may be even more problematic. Even in the simple realm of output fonts on printers being misinterpreted by input optical character recognition on input, the same output from one interface may produce different input on another interface. When passing through the physical world, digital data may no longer obey the laws of the digital world, and the result when reusing the inputs may therefore also disobey these laws. Even the most fundamental properties, like convergence, are no longer true at the interface.

Positive feedback in the physical space resulting from uncontrolled or not sensed parameters in digital space may produce FSM-induced physical hazards that the FSM may not record or control, and that the examiner may not be able to identify or speak to.

Reliability issues

Reliability is a fundamental concept to legal issues surrounding technical or scientific expertise and evidence. But there is little if any available information on reliability associated with digital space.

Faults, fault models, and reliability

A design approach widely considered reasonable, is to start with specifications and use them to produce designs. The designs are implemented and tested, and the resulting implementations are put into use, repaired, and updated over time. Faults can occur at each step of this process. If those faults get exposed in use, failures may result, in that the resulting outputs and states are not identical to those desired by the people who created the mechanisms.

The areas of fault tolerant computing, testing, reliability, and other related areas were formed as a result of a lack of in-depth understanding of these issues in digital systems hardware, and over a period of many years, these fields progressed toward a methodology for understanding and reducing faults and failures in digital systems. As a result, current digital systems have largely known hardware fault types and failure modes, they are tested at multiple phases of design and implementation, they contain special

test mechanisms to allow them to be tested at different phases of their lifecycles, and the processes used to make them are improved over time to reliability levels that are well defined and extraordinary in comparison to anything else we are aware of that is of similar complexity.

Fault models in the hardware space typically include, without limit, stuck-at faults (i.e., a bit is stuck at 1 or 0), bridging faults (i.e., two components that are supposed to be isolated are connected), open and closed circuit faults (i.e., a circuit is connected when it should not be or not connected when it should be), pattern sensitive faults (i.e., electromagnetic conditions when particular patterns of state and input are present cause an undesired state or output change), and transient faults (i.e., conditions such as solar flares, excessive temperatures, or other mechanisms cause a state or output to be improperly set, but when the conditions are removed, normal operation continues). Some systems are even designed to "fail safe" (i.e., in a "safe" mode) under particular fault conditions.

Measures of faults are made using tests that are generated in such a way as to reveal known portions of the possible faults under various fault models. These measures are made to defined coverage levels (i.e., an identified portion of the known faults in the model are tested and results presented in terms of the coverage as the percentage of faults covered by the tests), and coverage can be traded off against time to test. In many cases designs are altered to allow higher coverage levels in tests or to introduce special test modes that expose internal states for examination so that more higher coverage tests can be done in less time. In some cases, complete tests (i.e., with 100% coverage) are done, and known failure rate characteristics with use are applied to define "burn-in" tests.

At the system level, power on self-test mechanisms are placed in hardware and power on self-test (POST) regimens are supported by computer system basic input output systems (BIOS) startup code of many modern systems.

Reliability statistics are generated in all phases of these processes, so that the design and manufacturing processes undergo ongoing testing and improvement. Manufacturing processes are watched

particularly closely to detect faults that reflect process problems, so that the manufacturing lines can be repaired and adjusted before such faults produce high cost losses. These may range from changes in temperature or humidity levels to adjustment of chemical mixes, recalibration of machine tools, changes in times spent in different process steps, and so forth. In many cases, items that are not defective, but that are not up to the highest quality standard are measured and used in less stressful conditions. For example, processor speed or operating temperature ranges may be set appropriate to the quality of the manufacturing process as measured,

All of these steps are taken as part of the normal process of manufacturing digital circuits, circuit boards, systems containing those mechanisms, and the various components that go into the delivered composite digital system as delivered. But similar methods and mechanisms do not exist in most of the software industry, and this is key to the forensic examiner.

Hardware errors and reliability

In the physical realm, digital systems are designed to meet and tested to confirm they meet specifications that are quite stringent. For example, the mean time to failure for most computer systems is in excess of 2 years, and much of the failure rate stems from the power supply, glue on heat sinks, and similar non-digital components. Disk error rates, once the various mechanisms used to compensate for them tend to be nearly zero for the first few years of operation, and coding used on storage media provides for single bit error correction, and double bit error correction in many cases. Published error rates for devices are often made publicly available, and testing is regularly performed to try to determine error rates based on operating parameters and sell devices with operating parameters specified appropriately for the desired expected life of the product.

Depending on the specifics of the physical errors, many such errors are caught during POST, and additional methods are used at operating system startup and shutdown to identify and try to compensate for errors producing inconsistent disk state, even if at the expense of sometimes losing some of the recently written

content. Many of these conditions produce error logs that are retained on media and can be checked by examiners to identify conditions likely to be associated with such errors.

Forensic processes typically use additional coding methods, such as cryptographic checksums, to verify that images of media are unaltered between the time they are collected and the time they are provided to the examiner for examination. While these methods cover the traces examined by examiners, and compensate for many hardware errors associated with imaging, storage, and transport, they do not cover most aspects of the hardware used by examiners when performing examinations.

Examiners should take hardware reliability issues into consideration during examination and in their reporting, to the extent that they are relevant to the matters at hand. They normally don't have to worry significantly about it as part of their examination process, however, they should do verification of hardware operation before and after examination as part of standard reliability and calibration methods.

Software errors and reliability

Unlike hardware, which is usually well characterized in terms of error rates and mechanisms, substantially tested to defined coverage levels during all phases of manufacturing, and in which many faults producing failures will be rapidly detected, the software situation is largely the opposite.

Very few software mechanisms are subjected to the same sort of quality control and testing regimen that hardware goes through. There are very few fault models applied to software, it is almost never characterized in terms of error rates or failure modes as delivered, testing is limited and to undefined coverage levels, and when it fails, the failures are often unnoticed and unreported.

This applies both to the systems that produced the traces being examined and the examination system itself. Common fault types in software include, without limit; off-by-one errors (i.e., a result or analytical method goes too far or not far enough by one step), overflows (i.e., the storage areas associated with something being stored is not big enough to hold the input sequence), differences in interpreting the meanings of specifications like “word”, “line”,

"character", etc. (e.g., when asking for words ending in "ing" this may not include words that go across line breaks and have hyphens), syntax errors (I.e., specifications that are different from the intent being specified), type mismatch errors (e.g., searching for a string in one coding when it is represented in another coding), algorithmic errors (i.e., the algorithm fails to take into account a condition that exists and was not identified and corrected in testing), language understanding errors (i.e., the programmer had a misunderstanding about semantics of some language construct that was not detected and repaired in testing), missed implementation or environment differences (e.g., something that works in one version of an operating environment but does not work the same way in another version and that is not compensated for by the programmer), failure to detect failures in the middle of a process (e.g., a failed program execution producing usable but incorrect results is passed to subsequent processing steps which seem to work correctly and yield results that fail to indicate the intermediate failure), and presentation faults (i.e., the presentation of results is not readily understood by the examiner).

While these examples are intended to be instructive, they are by no means comprehensive. Furthermore, the more complex a program and the less it is based on a well defined mathematical approach to solving a specific and limited problem, the more likely it is to have more of these and other sorts of faults. Software faults are sometimes indicated in terms of detected defects (D) per thousands of lines of code (KLOC), and rates on the order of 0.25 D/KLOC were found for widely examined open source programs in 2008.[91] The average number of defects per project (i.e., program) was 283.49, with the lowest being 1 and the highest being 4967. The detailed defects detected were: NULL pointer dereference, resource leak, unintentional ignored expressions, use before test (NULL), buffer overrun (statically allocated), use after free, unsafe use of returned NULL, uninitialized values read, unsafe use of returned negative, type and allocation size mismatch, buffer overrun (dynamically allocated), and use before test (negative). These are all highly technical sorts of defects typically associated

91 "Open Source Report", http://scan.coverity.com/report/Coverity_White_Paper-Scan_Open_Source_Report_2008.pdf, 2008.

with security-related issues, and represent only a small portion of the overall fault landscape.

Since not even one program examined even with this limited set of fault typed identified was without defect, and because this represents only a small portion of the overall set of programs operating within normal operating environments, examiners cannot count on FSMs they use to do their work or used in systems generating traces they examine to always produce what their documentation documents, designers claim, implementers assert, or users observe in other instances.

Reliability of software is limited and, in most cases, unknown for the software in use. No level of user-level “black-box” testing can reasonably be expected to reveal the sorts of faults detected by the “white box” testing used in typical studies, and details of fault levels are rarely available for software used by most examiners most of the time. There is no widely accepted fault model for overall digital systems at the level of full scale typical system behaviors, but there are models at the hardware level[92] and some models have been applied at higher levels with limited adoption and limited success.

Cognitive limits of computers and people constructing them

The issue of fault models drives toward another fundamental difference between digital systems and people. Humans have cognitive limits associated with the way brains and bodies work. There are well known and widely published sets of cognitive errors that humans make,[93,94,95,96] and these sorts of errors are often made when humans design and implement digital systems. Certain types of errors recur, and while the community that designs systems works to reduce or eliminate the most common ones, such as off-

92 M. Breuer, A. Friedman, "Diagnosis and Reliable Design of Digital Systems", M. A. Breuer and A. D. Friedman, Computer Science Press, 1981, Breuer, Rockville, Md.

93 Bob Fellows, "Easily Fooled", Mind Matters, PO Box 16557, Minneapolis, MN 55416, 2000.

94 Thomas Gilovich, "How We Know What Isn't So: The fallibility of human reason in everyday life", Free Press, NY, 1991.

95 Charles K. West, "The Social and Psychological Distortion of Information", Nelson-Hall, Chicago, 1981.

96 Robert B. Cialdini, "Influence: Science and Practice", Allyn and Bacon, Boston, 2001.

by-one errors and failures to check input syntax, these and others remain today as they have for many years.

Reliability and its impacts

The impact of the limits on reliability for examination software and systems producing traces under examination are fairly clear from a standpoint of digital physics.

Software, which is the dominant source of FSMs producing traces today, has substantial numbers of identifiable faults, is largely black box to the examiner, and because of legal restraints, reverse engineering is not permitted for digital forensics in the US today.[97]

The effect of a fault not known to the examiner is that an unknown and potentially undesired state, state change, or output may occur at any time, potentially producing or failing to produce an otherwise unidentified trace that is inconsistent with the normal functioning of the FSM.

Unlike hardware faults which are typically exercised quite often and thus reveal themselves through behaviors, software faults are, in some cases, rarely exercised and thus harder to identify in testing and with side effects that may not be observed in reconstruction.

The effect of software faults on FSM execution over time is unclear, and hard to determine. While the general principals of digital physics will continue to apply, the certainty with which statements may be made regarding the sources of traces is reduced by the reliability issues in software, and the inability to characterize software precisely in this regard is potentially problematic for making statements about the reliability of the methodologies applied by the examiner.

Some legal perspectives

From the legal point of view, some of the implications of the nature of digital space have impacts on the manner in which DFE must be applied.

97 F. Cohen, “The DMCA Still Restricts Forensics”, 2010-08, available at http://all.net/Analyst/2010-08.pdf

Forgery is indiscernible at the level of individual bits

An important side effect of the artificial nature of DFE is that it can be readily forged, in the sense that any particular sequence of bits can be generated by any number of different mechanisms, and the resulting sequences generated may be indistinguishable from any original writing. Thus special attention must be paid to the larger situation in order to prevent a forgery or poor fidelity copy from being misinterpreted as original writing or its equivalent.

DFE is latent by nature so reliable tools must be used

While there are examples of digital data that can be physically seen or otherwise observed by people with their own senses, the vast majority of DFE is latent, in that tools are required in order to observe it. Perhaps even more importantly, the large volume of bits commonly present in legal matters leads to the need to use automated tools to analyze the evidence, and presentations of the evidence in many cases applies only representative samples and not an exhaustive presentation of each of the items of interest.

For this reason, tools are almost always relied upon to observe the evidence, to analyze and characterize it, to present it, and to draw conclusions about it. This means that the validity of the tools are very important to the validity of the evidence and the analytical processes used by the examiner.

There are also many different ways to visualize and present digital content, and the courts have not decided and likely will not decide, on an *a priori* basis, which presentation method is more probative than prejudicial or preferred over which other in which circumstances. Since essentially all such presentations use tools to turn bit sequences into things that people can observe with their own senses, the validity of the presentations and the tools that produce those presentations must meet the legal standard, which under the FRE[98] (702) call for a reliable method reliably applied. In light of the reliability issues discussed earlier, this is a substantial challenge.

98 The U.S. Federal Rules of Evidence.

DFE is trace evidence but not transfer evidence

While most physical evidence is understood in terms of a theory based on the transfer of divisible pieces of matter between things that come into proximity of each other,[99] this is not the way the physics of digital information works. The process by which each physical object coming into contact with another physical object leaves part of itself with the other physical object is called transfer, and the parts each leaves with the other as well as the missing parts resulting from such contact are called traces, and they are traces of events that took place in the physical world.

In the digital world, traces are the results of the result of the storage of bit sequences resulting from FSM operations. Each mechanism from which stored states are retrieved or collection mechanism that examines states and stores resulting bit sequences, produces its own traces in the form of those bit sequences. Digital systems that communicate may each, as a part of their independent operations, generate and store traces. Thus the mechanisms of generating a trace in a digital system is different from those for physical evidence produced through transfer, and the traces produced by different FSMs from the same event sequence may be completely different.

DFE admission is still complex and unsettled

DFE cannot stand up in court, point a finger at a party, and state that this person did that thing. In fact, there is almost certainly no case in which an expert who is applying sound scientific principles to DFE can state that, based on the DFE alone, and in light of their expertise and experience, this person did that thing.

> ***DFE does not and cannot, on its own, definitively place a person at a place, demonstrate that any specific physical act took place, or prove that a certain thing happened at a certain time.***
>
> ***Rather, DFE can only really be used to show that certain event sequences are or are not consistent with the available traces and to show that certain traces do, can, do not, or can not result from certain event sequences.***

99 E. Locard, "The Analysis of Dust Traces", Revue International de Criminalistique I. #s 4-5, 1929, pp 176-249, (translated into English and reprinted in 3 parts in A, J. Police Science, 1930 in V1#3, May-Jun 1930, pp276-298, V1#4 Jul-Aug 1930, pp 401-418, and V1#5 Sep-Oct 1930, pp 496-514.)

Traces are reflective of a stored state (situation) after the fact (post facto). Given an incomplete trace, the event sequence producing the trace cannot be uniquely identified. But that does not mean that any event sequence can produce any trace. Rather, a constrained envelope of causes (event sequences) may lead to any given effect (trace). Thus they may be used to identify constraints as to the event sequence that took place, but almost never produce constraints that produce unique sequences at the level of bits.

It is also noteworthy that people who stand up in court, point their finger at a defendant, and state that the individual standing right in front of them perpetrated a particular crime, have unknowingly given false identifications, even when they were cognizant of the need to remember the individual at the time of the crime, had the foresight to try to recognize the perpetrator during the crime, and even when the crime involved face to face personal contact over the period of minutes or longer.

The hearsay nature of DFE is typically overcome by having an individual, knowledgeable in how the traces were created and retained, testify as to their authenticity and nature. The normal business records exception typically applies in cases where the traces are kept and used as a normal part of doing business and are relied upon for normal business activities. Other sorts of DFE may be more problematic, but historically, DFE has often been admitted, even when records were not kept as part of a business.

A lack of a sufficient chain of custody, failure to seek additional warrants under the plain sight doctrine, spoliation, fabrication, and alteration, have been used to disallow traces, but this is often difficult to achieve, and this is not a throughly settled part of law. For example, in a recent California 9th circuit case, search and seizure limits were abused by law enforcement, courts acted to constrain the behaviors, and those constraints were then slightly reduced.[100]

A wide range of legal rulings dealing with issues ranging from discovery to retention to reliability have started to emerge, and these rulings are reaching further into the appeals process with time. The lack of clarity surrounding the scientific basis for various aspects of the issues seems to be based on inadequate

100United States of America v. Comprehensive Drug Testing, Inc., Sep 13, 2010.

understanding both within the courts and within the digital forensics communities. For these reasons, many further advances in digital forensics depend on continuing to produce technical information on the physics of digital information.

Summary of properties

Table 3.1 is a summary of digital and physical properties discussed. The terminology and differences between the worlds is intended as definitional for the rest of the book. It is hoped that it will be used throughout the DFE examination community and elsewhere.

Digital World	Physical World
Finite time granularity (the clock)	Infinite time granularity
Finite space granularity (the bit)	Infinite space granularity
Observation without alteration	No observation w/out alteration
Exact copies, original intact	No exact copy, original changed
Theft without direct loss	Theft produces direct loss
Finite (fast) rate of movement	No locality (entanglement)
An artifice created by people	A reality regardless of people
Finite State Machines (FSMs)	Physics and field equations
Homing sequences may exist	No perfect repeatability
Forward time perfect prediction	Forward time non-unique
Backward time non-unique	Backward time unique
Digital space converges in time	Physical space diverges in time
The results are always bits	The results are always continua
Results are always "Exact"	Results never perfectly known
Time is a partial ordering	Time is real(location)
Errors accumulate	Errors are local
Representation limits accuracy	Reality is what it is
Precision may exceed accuracy	Precision is potentially infinite
Forgery can be perfect	Forgery cannot be perfect

Digital World	Physical World
DFE is almost always latent	Some evidence is latent
DFE is trace but not transfer	Traces comes from transfers
DFE is circumstantial	Evidence is circumstantial
DFE is hearsay	Evidence is physical
DFE cannot place a person at a place at a time	Evidence may put an individual at a place at a time
DFE can show consistency or inconsistency only	Evidence can show more than just consistency
Probability is dubious	Probability is often usable
Content has information density	No defined density limits
Content density variable	Content density not controlled
Content perfectly compressible	No perfect compression
Digital signatures, fingerprints, etc. generated from content	Body (phenome) generated from DNA (genome)
Content meaning is dictated by context	No universal theory of meaning but physicality exists regardless
Context tends to be global and dramatically changes meaning	Context tends to be local and incrementally changes meaning
FSMs come to a conclusion	Eats shoots and leaves
Cognitive limits from program	Cognitive limits from physiology
Hardware fault models from computer engineering	Hardware fault models from physics
Time and space tradeoffs known	Tradeoffs unclear
Near perfect virtualization and simulation possible	No virtualization
Many nearly or equivalent FSMs	The uncertainty principal
Undecidable problems	Nothing known as "unthinkable"
Computational complexity limits computations	No well understood limits on new ideas

Digital World	Physical World
Everything is decidable	Many things are not decidable
Consistency is guaranteed	Consistency is possible
Completeness is guaranteed	Completeness is possible
Consistency AND completeness	Consistency OR completeness
Time limits on achievable results	Time limits unknown
Complexity-based designs	Complexity not determinant
Fault tolerance by design	Normally not fault tolerant
Accidental assumption violations	Assumptions non-violable
Intentional assumption violations	Assumptions non-violable
Discontinuous space	Continuous space
Discontinuous time	Continuous time
Minor differences amplified near discontinuities	Differences retain fidelity
Major differences suppressed away from discontinuities	Differences retain fidelity
Identical use of an interface may produce different results	No such thing as identical, each thing is unique
Ordering may be reversed	Ordering subject to light time
Value sorts may be reversed	Value sorts remain consistent
Actuate-sensors loop errors	Interference based errors
Sensors/actuators limited in physical properties	All physical properties present

Table 3.1 - Summary of information physics

Table 3.1 and portions thereof will be used throughout the rest of this book to review and consider the issues of information physics as part of considering the specific issues in DFE examination. In practical use, it is helpful to consider specific issues in each case, and this table may be useful in that light.

Extensions of the physics

Traces come from the operation of digital systems. The more the examiner knows about the systems in use, the more clarity forms around the consistency and inconsistency of traces with the environment. This then leads to additional operational constraints.

Just as the physics of digital information in the general sense applies to digital systems overall, but is a specialization of the more general physics of the universe we live in, extensions to the physics of digital information specialized to different operating environments are feasible, and to the extent that they are developed, they can be used for those environments in the same way as the present physics is used for digital systems in general.

The DFE examiner uses knowledge, skills, experience, and the results of training and education to extend physics to the specifics of the case. In doing so, the examiner provides the means to understand consistency, inconsistency, uncertainty, errors, and what may or may not have taken place.

Chapter Summary

The physics of digital information has many facets and they are not intuitive relative to the notions we have on a day-to-day basis in dealing with the physical world. The introduction of information physics is a way to bridge the gap between the precise details of the mathematics and physics of the underlying mechanisms and the everyday way that people think about these issues. Newtonian physics is slightly less precise than the more detailed wave equations and quantum physics, and information physics is slightly less precise than the underlying mathematics of digital systems.

Given that the summary information regarding the physics of digital systems is imprecise, the examiner who wishes to be truly precise should understand the underlying mathematics and physics, and bring this to bear in applying these results. But on a day-to-day basis, information physics in its summary forms may be used to quickly check on the reasonability of claims and to think about the issues in a legal matter with more clarity than might otherwise be attained. And that is where its utility lies.

Throughout the remainder of the book, versions of Table 3.1 are used to bring up questions and focus consideration of issues in DFE examination. This approach to examining DFE is one that, as the examiner gets used to it, provides far more insight far more quickly, and allows the wheat to be rapidly separated from the chaff.

The key points of this discussion are; (1) the general concept that, using these theoretical notions, a great deal of information about the evidence as it applies to the case may be gleaned with a sound scientific and mathematical basis; (2) the physics of information is not the same as the physics usually used in criminalistics, and (3) be careful in characterizing results - assumptions may be wrong.

Questions

1. Pick one of entries from Table 3.1 and provide a compelling argument in less than 250 words that allows a lay person to understand it. Pick 3 more and do the same thing.
2. Suppose that a party to a legal matter makes the following claim. Provide an analysis of this claim in light of all of the elements of information physics described herein. Assume that all of the information used to make this claim is digital forensic evidence, and identify all of the ways in which this statement can be addressed using information physics.

   > "The evidence clearly shows that the user known as Bill entered the room before 9 PM, that nobody else was present in the room at that time, and that the user known as Bill deleted all of the files on the system he was logged into over the period of the next 20 minutes, including all of the hard drives and the tape backups that were present in the room."

3. Given the nature of the digital space and the physical space, explain how things about the physical space can be legally demonstrated through the digital space.
4. Given that cryptographic hash functions are derived from content rather than generative of it, and given that all such hashes are not unique to any given document, how can these hash functions be used to authenticate that content is not changed during examination?
5. Just because a problem has exponential complexity, that doesn't mean that it cannot be solved in analysis. Explain how the examiner can use their knowledge of complexity to make decisions about procedures they may undertake.
6. Identify three potential extensions to information physics that apply to an operating environment you are familiar with. Provide details on how these three extensions may be used in evaluating DFE that is asserted to exist within that environment, and provide additional guidance on the limits of the use of those extensions.

7. Identify three potential extensions to information physics that apply to a computer language you are familiar with and provide details on how these three extensions may be used in evaluating DFE that is asserted to exist within that language.
8. Assume that in every case worked, the items identified from information physics will be used as a basis for evaluating results, and that all examiners are aware of and understand all of these issues at a detailed level. What processes and methods does this imply, and how would you structure an examination methodology to take this into account?
9. Consider question 8 with the twist that one side in a legal matter understands and has studied information physics and the other side has not. How much of an advantage will one side have over the other, and what will likely be the outcome of this disparity in knowledge and approaches?
10. Suppose one of the elements of information physics described herein turns out to be wrong. How might this impact examination from here going forward, and why?
11. Imagine a system in which complete traces are generated and retained by an external “audit system”. Could such a system ever exist, and if it did, what sorts of properties would it necessarily have?
12. Describe an approach to defining the reliability of the methods used for examination given that those methods rely on mechanisms with the reliability properties identified with regard to software.
13. Suppose we completely abandon the digital physics approach and revert to the physics of the natural world. How will this impact digital forensics, and to what extent will it be usable to counter the physics of digital information as presented here?

4 A theoretical examination framework

A theoretical framework is used to frame some of the technical issues associated with digital forensics. In this model, detailed trajectories through the legal requirements associated with charges are associated with event chains, supported by traces that may be applied to multiple events, and costs and schedules are associated with processes.

Previous models

In recent years, there have been various attempts to model the digital forensic evidence examination process. These models range from reviews of methods in common use in different communities to more detailed theoretical models of how digital systems work. None of these models have been analyzed to date in light of the full set of issues associated with information physics, but two have been explored in some depth and with substantial rigor. These are the models of Carrier[101] and Gladyshev.[102] Each model was the subject of a doctoral dissertation in digital forensics, and each has its own features that are worth understanding. These will be reviewed in the order of their publication.

Models of processing evidence and making choices about what to process and when to stop have also been undertaken, with the most relevant one to our discussion being the one of Kwan et. al.[103] One of the central features of the Gladyshev and Carrier models is that they assume that the detailed definitions of all relevant FSMs are known to the level of granularity of the model. This is fundamentally problematic in real legal matters because of the nature of how evidence is made available to experts doing examinations. They both also largely ignore the issues of time and complexity in the examination and legal processes that limit the practical applicability of their models. That is not to say that they are

101B. Carrier, "A Hypothesis Based Approach to Digital Forensic Investigation." PhD Dissertation; Purdue University; May, 2006.

102 P. Gladyshev, "Formalising event reconstruction in digital investigations." PhD Dissertation; University College Dublin; 2004-08.

103 M Kwan, K P Chow, F Law & P Lai, "Reasoning About Evidence Using Bayesian Networks", Advances in Digital Forensics IV, 2008, pp.141-155.

not worthwhile and cannot be used in practice, but rather that they lack in some of the areas where Kwan et. al. provide insight.

Gladyshev's model

This mode[104] is about formalizing the reconstruction of event sequences associated with digital forensic investigations. It asserts, in essence, that "the system under investigation is modeled as a finite state machine. The available evidence is modeled as the evidential statement, which expresses the evidence as a collection of witness observation about the state and change of observable system properties during the incident. The event reconstruction problem is then defined as finding all possible explanations for the given evidential statement with respect to the given finite state machine."[105] "Possible sequences of events that could have happened in the system during the incident can be determined by (1) backtracing transitions leading to the final state x; and (2) discarding sequences of transitions that disagree with the available evidence."

This model assumes that the observer can see some number of time pictures of the state of the finite state machine (FSM), and "partitioned runs" consist of the sequences of events between known states with "runs" comprising the collection of partitioned runs concatenated together. Evidence outside of the digital system is considered as sets of triples (P, M, O) where P is the set of all computations that possess the properties observed by the witness and M and O bound the earliest and latest time these properties were known to hold. Observations are strictly bounded to the set of states of the FSM, its inputs, and outputs, and thus the discussion surrounds the activities of the FSM and not the physical world in which it operates. The reconstruction problem is then to create all possible runs of the FSM that produce all of the observations.

In subsequent papers, traces such as time stamps, file system state, and other related information are formalized into this structure and theoretical results. For example, the fact that one

104 P. Gladyshev, "Formalising event reconstruction in digital investigations." PhD Dissertation; University College Dublin; 2004-08.

105 P. Gladyshev, "Adding real time into state machine analysis of digital evidence", Technical Report UCD-CSI-2006-3, 2006, School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland.

thing must have happened before another, is applied to confirm or refute possible runs. The actual analysis is performed at a higher level than the FSMs in most of the subsequent works, but the approach is fundamentally sound in the sense that, if all possible consistent runs are identified, either the observations are wrong, the model is wrong, or one of those runs must have taken place.

Carrier's model

This model[106] takes the general approach of identifying machine histories. This model formally defines an investigation and set of techniques based on an extended FSM model that adds removable devices and more complex states and events, but that is still reducible to FSMs and thus consistent with previous works. Carrier was aware of Gladyshev's work and cited relevant papers. Carrier rightly recognizes that "This process requires that Q, Σ, and δ be fully understood and therefore is used only with small systems, such as slack space of a file ... and printer queues (where only high-level complex events are considered)." Carrier also identifies "...there are many differences. The biggest is that the history model does not require that the system be modeled as a FSM for an investigation to occur ... and it can be used to identify what assumptions are being made during an investigation."

Carrier states that "This work defines a model that can describe the previous events and states of a computer at the primitive and abstract levels [and] uses the model to define 31 unique classes of analysis techniques, ... organized into seven categories. Completeness for the categories can be shown ..." The real advantage to this approach is that the explicit use of abstract levels provides a mechanism for dealing with the complexity issues of the detailed FSMs, which Carrier calls the "primitive history". This is something the Gladyshev ultimately deals with as well, but not as explicitly. "Complex events" are defined along with complex storage and complex states and thus a complex version of the FSM is invoked as consisting of inputs (I), States (S), Outputs (O), and maps from IxS to S and S', as in the FSM model. In essence, this approach creates equivalence classes between sets of primitives and their complex versions. Histories are then defined in terms of

106 B. Carrier, "A Hypothesis Based Approach to Digital Forensic Investigation." PhD Dissertation; Purdue University; May, 2006.

these class sets, and most of the same sorts of things identified by Gladyshev should apply, subject to some limitations.

In particular, the underlying assumptions lead to results not being as precise or accurate as in the case of using the bit-level reality that information physics applies to. With these assumptions, it may be that some of the information physics properties no longer hold for these class sets, and if that is true, there may be many classes of consistencies and inconsistencies at the information physics level that cannot be differentiated in the class sets.

Another problem with this approach is that there are many possible class sets that can be defined, and the class sets chosen are chosen by the investigator(s) to seek to determine the issues that they are investigating. Of course two investigators might choose different class sets, in which case they could have dramatically different results. Carrier asserts that these class sets are defined based on the observations and hypotheses in the case, but how this is done from an actual case remains problematic both then and now. The examination process based on the scientific method is outlined as:

1. Observation: Relevant information is collected/observed.
2. Hypothesis: Observations drive hypotheses formulation.
3. Prediction: Predictions about evidence are made.
4. Testing: Predictions are tested against evidence.

Carrier also defines 7 categories of analysis techniques, which I will compress into a logical sentence as:

{primitive/complex}x{event/state} at relevant times

Hypotheses in terms of sets of possible events that transition from state to state are made and compared to observations at relevant times. If consistent, the hypothesis is confirmed, and if not, it is refuted. Unless the space of all relevant hypotheses is covered, no number of confirmations constitutes proof, but a single refutation constitutes disproof. Hypotheses may involve complex chains of using tools for indirect observations, and hypotheses may be refuted because of process faults or other assumptions as well as inconsistency of the actual issues in the case.

Kwan et. al.'s model

The approach taken in Kwan et. al.[107] may be characterized as:

1. A legal requirement (L:{l_1, ..., l_n}) associated with a violation (V) consisting of the union of a set of circumstances such that each circumstance must be shown true to within the standard of proof in order to warrant the charge of a violation based on the defined legal criteria.
2. For each element of the legal requirement [$\forall l \in L$] there is a set of evidence chains *E:* {E_1, ..., E_o}], each of which consists of a set of events (e) evidenced by any of a set of traces T: {t_1, ..., t_n} of those events within the digital system [$\forall E_x \in E, E_x : \exists \{e_{x1}, ..., e_{xp}\}, \forall e_{ab} \in Ex, \exists t_c \in T : t_c \rightarrow e_{ab}$].
3. Each item of evidence has an assumed weight $W_x=(w_{x1}, ..., w_{xp})$ normalized to a total weight of 1, so that $\sum(w_{x1}, ..., w_{xp}) =1$, and a cost of detection c_{xa} so that to total cost of detection for any given chain of evidence is fixed $C_x=\sum(c_{x1}, ..., c_{xp})$.
4. An investigation starts in phase 1, and as the investigation proceeds, each item of evidence detected contributes to the weight and each effort to detect evidence contributes to the cost. If W exceeds an organizationally defined threshold of adequacy (g), the investigation goes to phase 2. If W gets low enough that the total available weight of evidence left to detect cannot reach g, the investigation is abandoned.
5. In phase 2, a Bayesian network is used to analyze the evidence against a hypothesis of how the crime was committed. This network uses *a priori* probabilities of traces indicating guilt and yields a probability of guilt (G). When $\forall l \in L, \exists E_x : P_x > g_x$, G is adequately established to propose charges. In[108] G is calculated as the product of the *a priori* probabilities. For example, the presence of a known Trojan

107 M Kwan, K P Chow, F Law & P Lai, "Reasoning About Evidence Using Bayesian Networks", Advances in Digital Forensics IV, 2008, pp.141-155.

108 R. Overill, M. Kwan, K. Chow, P. Lai, and F. Law, "A Cost-Effective Forensic Investigation Model", IFIP WG 11.9, International Conference on Digital Forensics, Jan 25-27, 2009.

can be established with a probability of approximately 0.98, if the claims of anti-virus vendors can be believed.

Legal precedent provides well-established subsets of the overall structure, so that the full complexity of the space is not normally exercised. In particular, once a successful prosecution is made, the set of evidence required for the particular path through the structure is established and the same elements may be repeated with greater certainty of success in court. This increases the weight of the elements of that particular path. In addition, the prior development of methods to establish that path through the structure can be reapplied, thus reducing the cost of using the same methods to make future cases. The strategy applied in[109] was to use an existing path through the structure based on precedence and identify the lowest cost element of the evidence sequence for each step in detection and analysis. In this way, if a required element is not found, a lower cost will have been expended prior to determining that the path is infeasible, and the most expensive elements to detect are delayed until the less expensive elements are detected. There is also an implicit assumption in this model that elements are independent, costs are independent, and benefits do not accrue across multiple paths. In effect, multiple paths are not typically taken in this approach because the overall value of detecting any particular criminal committing any particular crime is not normally high enough to justify complex examination. There are plenty of potential crimes and evidence available to consume available resources, and resource minimization with conviction maximization is the goal.

The present model

The model described herein consists of a legal context, (L, R, V) with a set of hypothesized claims (H) supported by sets of events (E). Traces (T) from the digital forensic evidence are analyzed based on internal consistency (C) and consistency with events (D) through the use of forensic procedures (P) using available resources ($\mathcal{R}$) within a schedule (S). The result of examination is a

109 M Kwan, K P Chow, F Law & P Lai, "Reasoning About Evidence Using Bayesian Networks", Advances in Digital Forensics IV, 2008, pp.141-155.

set of facts that tend to support (+1), refute (-1), or are orthogonal to (0) the issues in the case.

The legal context

A legal statute, or law (L) is associated with a violation (V), using a logic expression L:{l_1, ..., l_n}, R:{r_1, ..., r_m}, LxR→[F|T], where l_x is an element of the statute and R is a relationship between elements of the statute so that, if the set of elements required to meet the relationship defining a violation (the truth of LxR) are present, it implies that a charge of violation is warranted based on the defined legal criteria. (LxR→V) For example, the CAN-SPAM act[110] is a US Federal statute that reads, in part, "(a) Whoever, [for commerce] knowingly ... (3) materially falsifies header information in multiple commercial electronic mail messages and intentionally initiates the transmission of such messages...shall be punished...". This statute (L) can then be broken down into elements including (l_1) the act was for commercial purposes, (l_2) there is material falsification of a header, (l_3) the falsification is present in more than one email message, (l_4) the actor initiated the transmission of these messages, and (l_5) that initiation was the intent of the actor. All of these must be proven to within the standard of proof by the charging party in order for the punishment to be invoked, so the resulting expression might be of the form L=(l_1*l_2*l_3*l_4*l_5).

The hypothesized claims

Claims, which we will call hypotheses, *H*={H_1, ..., H_n} are made by one party or the other in the form of statements which may be supported or refuted by digital forensic evidence and which tend to support or refute the violation. For example, (H_1) Defendant sent email messages accompanied by falsified, misrepresented, or forged header information and (H_2) Defendant sent or caused to be sent at least 20,000 false and/or deceptive commercial e-mail advertisements to Plaintiff's servers.

The hypothesized events

For each element of the legal requirement ∀l∈L there is a set of event claims *E:* {E_1, ..., E_o}; each of which consists of a set of

110 15 USC 103 "Controlling the Assault of Non-Solicited Pornography and Marketing Act of 2003", or the "CAN-SPAM Act of 2003", http://uscode.house.gov/download/pls/15C103.txt

indicated events from the set of all events [$\forall e, e \in E^*$] within and outside of the digital system $\forall E_x \in E$, $E_x:(e_{x1} \in E^*, ..., e_{xp} \in E^*)$; that, when put together, purport to constitute a demonstration that the relevant legal requirement is met. Again, from the example, (E_1) Emails received by Plaintiff contained or were accompanied by falsified, misrepresented, and/or forged header information, (E_2), Emails received by Plaintiff had subject lines designed to mislead a recipient regarding the contents or subject matter of a message, (E_3) Emails received by Plaintiff were sent by Defendant and/or their agents, (E_4) These emails contained false, misrepresented, and/or forged header information, etc. As events; (e_a) The “HELO” protocols on some of the emails provided “identities” of the sending computers that do not match the IP addresses of the sending computers; (e_b) The “identities” provided by Defendant and/or its agents or the computers delivering the emails do not match the IP addresses of the contacting computers; (e_c) These “identities” not matching IP addresses constitutes falsified, misrepresented, and/or forged information contained within a header; and (e_d) The failure of a lookup used by Plaintiff to try to match a domain name to an IP address or an IP address to a domain name to so match indicates a willful act of of falsehood, intent to misrepresent, or forgery, and so forth. Such claims may or may not be reasonable, logical, or internally consistent in terms of the things they purport to assert, and may include assertions made by counsel, asserted facts, or statements from other sources, including those of the parties involved. The DFE examiner's challenge is to confirm or refute these events.

The traces

There is the set of possible digital traces from existing evidence [T: (t_1, ...,t_q)], each element of which may exist. T may be incomplete in that $\exists t:t \notin T$. Subsets [$\tau \subseteq T$] tend to support or refute events relevant to the matter at hand. In the example used here, only a small subset of the asserted events can be confirmed or refuted by DFE, and in many cases, only elements of those events produce traces indicative of those events. For example, there may be classes (c) of traces of events within email headers, including, without limit:

(c_1) Date and time stamps in "Received:" headers,

(c_2) IP addresses in "Received:" headers,

(c_3) Domain names in "Received:" headers,

(c_4) Formatting of "Received:" headers,

(c_5) Presence of "Received:" headers from locations,

(c_6) Spelling of header names, and

(c_7) Message identifiers

and so forth. There are clearly large numbers of such traces and many ways to combine and compare traces to other traces.

The internal consistency relationship between traces

There is an internal consistency relation $C:TxT \rightarrow [-1...1]$ between traces, represented as C, $\forall c \in C$, $c \rightarrow [-1...1]$. All sets of traces relate to all other sets of traces by a value ranging from -1 (completely inconsistent) to 1 (completely consistent), with a 0 value indicating that the relationship is not revealing. For example, and without limit:

(1) Lack of correspondence between dates and times in "From " separators to "Received:" headers,

(2) Sequences of times in "From " separators within mailbox files that are not in time sequence,

(3) Mismatches of "From " separators with other content,

(4) Different "From " separators with identical headers and/or bodies, and

(5) Identical "From " separators with differing headers and/or bodies

are all inconsistencies consistent with fabrication, alteration, or spoliation. At a minimum, such a trace is not consistent with original writing from a mail transfer agent (MTA). ***WARNING:*** *not all apparent inconsistencies are what they seem to be. For example, time stamps may represent different clock settings or similar things.*

The demonstration consistency of traces

There is another consistency relation $D:TxE^* \rightarrow [-1..1]$, called demonstration consistency, that relates all possible traces T to all possible sets of identified events E*, and which may tend to confirm

or refute hypothesized sets of events by ranging from event sets being completely inconsistent with traces (-1) to event sets being completely consistent with traces (1), with 0 valued demonstrations indicating that the demonstration is not revealing. For example:

(1) From "(c_2) IP addresses from 'Received:' headers"; if an email that is claimed as a violation by Plaintiff has a trace indicative of an IP address of a competitor of Defendant, "(e_h) Plaintiff received commercial email messages sent by Defendant and/or their agents" would seem to be refuted for that email by that trace.

(2) Date and time stamps in "Received:" headers either indicate that emails were or were not sent within time frames limited by the statute of limitations of the applicable law. Given that the CAN-SPAM act was enacted after 2000, an instance of an email with a "Received: " from header indicative of a date prior to 2000 would appear to be inconsistent with a violation of the CAN SPAM Act, even if the implicit statutory dates are not identified by Plaintiff.

There may be many relations between traces found and events asserted, so the relationship of traces to events is a many to many onto relation. The presence of traces does not necessarily imply that those traces are reliable indicators. For example, computer dates may be incorrectly set and emails may be forged. The strength of a refutation depends on the accuracy of the dates in the headers, so additional relationships, such as the use of an anchor event in conjunction with another event[111] may result in a higher valued relation. Thus there is a synergistic effect between elements in subsets of D so that the combination of several traces may cause a far different evidential weight than the sum or the product of the individual weights.

The forensic procedures

There is a finite set of forensic procedures $P:\{p_1, ..., p_n\}$, $\forall p \in P$, $p \rightarrow \{c \subset C, \mathbb{D} \subset D, \mathbb{C} \not\subset C, \mathbb{D} \not\subset D\}$ available to the DFE examiner at any point in time. Procedures are normally implemented using methods and tools that have some properties. Each procedure has the potential

111 F. Cohen, "Issues and a case study in bulk email forensics", Fifth Annual IFIP WG 11.9 International Conference on Digital Forensics, Orlando, Florida, USA, January 25 - 28, 2009, also appearing in "Advances in Digital Forensics V".

to act on any subset of T and to produce false positives (makes), false negatives (misses), or to correctly find the presence or absence of subsets of C and/or D. For example, the use of the program "grep" in a Unix-like operating environment may be applied to traces to seek instances of strings typical of IP addresses within areas of traces typically associated with "Received:" headers. The presence of particular IP addresses identified as belonging to the Defendant may tend to support a particular event. However, the "grep" command may or may not be applied in such a manner as to produce false positives or false negatives, and thus it may make or miss connections between the traces it is applied to and relevant events. While legally, in most jurisdictions, all procedures are theoretically available to all parties, some procedures, either because they are not published, are prohibited, or because the examiner is unaware of them, may not be known to or available to any or all parties at any or all times.

Available resources

Parties have finite resources $\mathcal{R}:(\mathcal{T},\$,\mathcal{C},\mathcal{E})$. Procedures take time, money, capabilities, and expertise; and each of these elements limit the ability of the parties to fully examine the space of possibilities. A simple model of cost has been used to represent resource constraints,[112] but in general, the resource problem in digital forensics corresponds to the resource problem in other fields of optimization, and is usually an NP-complete or harder problem.

The schedule

A schedule sequence [$S:(s_1, s_2, ...),\forall s\in S$, $s:(L\subset L, R\subset R, H\subset H, E\subset E, T\subset T, C\subset C, D\subset D, P\subset P, \mathcal{R}\subset\mathcal{R}, t, t')$] exists where *t* and *t′* bound the time period for portions of the schedule, and only subsets of L, R, H, E, T, C, D, P, and $\mathcal{R}$ are available within that time frame. Arguments asserting and refuting claims are made to triers of fact (judges or juries) in a sequential fashion, with one side presenting then the other. A limited number of "rounds" of presentations are available, specific time frames and similar constraints are placed on all such information exchanges, and the standards of proof, ability of the triers of fact to understand the arguments, and space available for

112 R. Overill, M. Kwan, K. Chow, P. Lai, and F. Law, "A Cost-Effective Forensic Investigation Model", IFIP

presentation of arguments and facts vary with case type, jurisdiction, triers of fact, and situation. For example, without limit:

- Complaints may be amended over time.
- New evidence may be found or provided.
- Evidence may be excluded.
- Rulings, agreements, stipulations, or other interactions may effect which arguments are available.
- Theories of the case may change.
- Charges may be added or dropped.
- Events and event chains may be added or removed.
- Traces and classes of traces may be unavailable and become available, or available traces may become unavailable based on arguments, time, computational resources, or other factors.
- Internal consistencies or inconsistencies may be found or refuted.
- Demonstrations may become available or gain or lose strength over time.
- Procedures may become available due to research and development or become unavailable due to legal rulings or other external factors.

Procedures also consume resources that may or may not be within the capabilities of the examiners working for any given side on any given matter. For example:

- A change of venue might create a new schedule and increase the available time, making alternative procedures available.
- Financial limitations of a client might limit resources so as to reduce the amount of effort available and thus limit the available procedures.
- New inconsistent traces or events might arise, causing a new focus on these issues when time is limited.

- A deadline for an evidentiary hearing might force a focus of resources on a particular facet of the case forcing other facets to be set aside for a period.
- Confirming traces or events might arise in the middle of a case, causing a whole area of examination to be made invalid.
- After a particular phase of the legal proceedings is completed, there may be new traces found that show something that can no longer be legally demonstrated, such as a discovery that a piece of evidence was forged after the evidentiary hearing is closed and the evidence has been shown to the jury.

The number of possibilities is clearly large, and the impact on the matter may be profound.

Even though, from a logical standpoint, adequate confirmations or refutations may exist to secure a theoretical confirmation or refutation of charges, the actual legal matter may have an outcome that is inconsistent with the result of the logical analysis. For these reasons, a single confirmation or refutation is generally considered inadequate and, especially when a great deal is at stake and the participants have adequate resources to do so, more complete exploration of the space is undertaken. All of these impact the ability to and order of the search of the space of *T* and *E* and the search for relations C and D, and this affects the schedule. In cases where digital forensic issues are important, the potential consequences are high, and adequate resources are made available, a larger portion of the space of {L, R, H, E, T, C, D, P, $\mathcal{R}$} is likely to be explored.

Some discussion of the model

Within this model, which we believe encompasses the essential aspects of interest, certain things are clear.

The model is complicated

Depending on the nature of the challenge being met, different subsets of the proposed model may be applied and specific assumptions stated, with those now stated assumptions being

made clear by the selection of the subset of model elements. For example, the model of[113] can be seen in the context of this model to ignore all but a single element of S, and for that element, to ignore all but a single subclass of R, assume consistent E, ignore the details of T, C, and P, assume a single metric for D, and attend only to $ within $\mathcal{R}$.

The sizes of the model components

The size of the search space involved in any substantial matter is enormous, and thoroughly searching it or achieving substantial coverage of it, for any nontrivial matter, is infeasible. Specifically,

- L is finite, and for any given matter, it is defined by the specific laws.
- R is typically simple and is almost always expressible as a boolean function, perhaps with some metrics such as monetary thresholds.
- H is unlimited in possible makeup, but in any particular case, the elements of H get defined by documents provided by each side, and the courts prevent ongoing alteration H beyond some time within the schedule.
- E can be very large, but in most cases it is provided as a few hundred to a few thousand relevant events that are asserted, including statements made by the parties in depositions, testimony, and elsewhere.
- T, in its totality, is the size of all sets of all states of all digital automata in existence at all relevant times. But in any particular matter, T is limited to the traces collected. The size of even this reduced T is also very large, given that every possible subset of bits within all available DFE can constitute a trace. To get a sense of this, for a total trace of 8 bits, there are 2^8 different possible sets of bits that can comprise traces, and for each of those sets of bits, there are 2^n different possible traces (trace values of that set of bits), where n is the number of bits in the trace. For 8 bits, there is one combination of 8 bits forming an 8-bit trace, and there are

113 R. Overill, M. Kwan, K. Chow, P. Lai, and F. Law, "A Cost-Effective Forensic Investigation Model", IFIP

256 possible traces of that size. There are 8 different sets of bits comprising a trace of 7 bits, and each of those traces have 128 possible values (2^7), so there are $8*2^7$ different 7-bit traces, or 1024 of them. More generally, there are m!n sets of bits of length n in a collection of m bits, and for each of those, there are 2^n different possible traces. The total number of traces for m bits of data is then $\sum(m!n)2^n$ for n=1 to m. So the set of all possible traces for a single byte comes to $\sum(8!n)2^n$ for n=1 to 8, or 6560 unique 8-bit traces. For 16 bits, this comes to 43046720, and for 64 bits, it comes to 343368382029251248465784908928O, more than $3*10^{31}$. Clearly, for any substantial set of bits, the space of traces cannot be exhausted. The evidence identification problem is fundamentally about identifying relevant subsets of T, and this problem is not even close to being solved. However, legal precedent in the United States has led to the requirement to preserve evidence that might reasonably be believed to be relevant to the matter at hand, as of the time that any party has or reasonably should have knowledge that the evidence may be material. Thus the parties have an obligation to diligently identify and preserve, or cause to be preserved, traces like audit trails from contractors and providers, content from related systems, and any other such traces.

- C is the size of T squared, $[|T|^2]$, For substantial sized T, this is very large. For 64 bits of total evidence, the size of the set of all internal trace consistencies and inconsistencies is approximately 10^{63}. This makes any notion of coverage of C by exhaustion ridiculous on its face. It appears that a large portion of traces are independent of each other, but there may be any number of subtle interactions between traces. For example, a time stamp of user data entry to a database on one computer may be impacted by a Web page lookup on a seemingly unrelated computer. The deviation of timing of the data entry could be caused by a domain name system (DNS) lookup by the database engine delayed due to the DNS lookup associated with the Web page lookup on the seemingly unrelated computer, which has a trace in a Web server log. While finding and associating such a trace may

seem nearly impossible, it seems clear that in the interconnected world of the Internet, subtle effects exist and may leave traces. In practice, a relatively small set of traces may be examined, and the selection of the traces to be examined and method for doing such examination is not well defined or developed except in specific areas.

- D is the size of T times the size of the power set of E. That is, each subset of traces may interact with each subset of claimed events. This is again too large for practical exhaustion for any practical situation. Just as for C, there may be subtle interactions between distant traces and asserted events, and subtle effects may leave relations that are hard to identify. This goes to the problem of trace identification and collection as well as to analysis. As with C, a relatively small set of traces may be examined for a subset of the event sets, and the selection of the traces to be examined for event sets and method for doing such examination is not well defined or developed except in specific areas.
- P is at most the size of all possible instruction sequences executed on all subsets of T and E from all possible initial memory states, over a defined time. This is on the order of the number of different instructions in the processor (|i|) taken to the power of the instruction execution rate (r) times the available time (t). (i.e., $|i|^{tr}$) For an instruction set with 100 instructions executing at 10^9 instructions per second for one second, the number of different instruction sequences comes to a number written as approximately a 1 followed by 10^{18} 0s. This is then multiplied by the number of possible initial memory states and by the size of D to get the number of possible analyses that can be done in one second of computer time. The number of possible procedures is thus too large to contemplate, and actual procedures executed cover a very small subset of the total possible procedures. In practice, the number of actual procedures available is very small, being limited to the number of procedures developed by people or their machines, and the number of procedures that meet the legal requirements of being scientific according

to a defined methodology properly applied, being executed by tools that have been tested, calibrated, demonstrated to be reliable, and properly apply the defined methodology. There are, perhaps, a few thousand procedures that are so defined in digital forensics today, and the number that have been published and peer reviewed is smaller still.

- R acts to constrain the process in several dimensions. Constraints on time spent in examination are typically forced by a combination of the schedule, the actions of the parties and their legal teams, and limits on costs. Constraints on costs are dictated by the parties being represented and their available resources dedicated to the particular effort. For low-valued cases, little examination is likely, while for civil cases with many millions of dollars at stake or criminal cases that are "high profile", far greater costs are likely to be expended. DFE examination is only a small part of most overall cases, and thus it is usually only a small part of the cost involved in the overall matter. Capabilities of organizations like government agencies and large-scale corporate data centers are typically far greater than those of smaller firms and individual examiners. While rental forensic capabilities are starting to appear in the market, they tend to offer predominantly computing power, storage, and standard forensic search types of capabilities. Expertise is a far more difficult and expensive resource because it involves; people with knowledge, skills, training, experience, and education; that are able to combine understanding of the legal situation with understanding of technology, computer programming, and operations; to create analytical methods that both meet the needs of the legal system and are revealing with respect to the matter at hand. Given the relatively small number of publications in this arena and the small number of experts participating in open professional societies, human expertise may be the most constrained resource in nontrivial matters.
- S acts to constrain the process in real-time and alters the nature of the forensics effort over time, sometimes quite dramatically. Depending on the specifics of the legal matter, the total time frame from first notice of a legal matter to final

disposition may be as short as a few weeks or as long as tens of years. Typical matters are resolved in less than two years, and deadlines are commonly on the order of weeks apart.

Limits on what we know about this model and digital forensics

While this model is intended to depict the nature of the legal system as it applies digital forensics and the inherent nature of trace evidence as it applies to legal matters in the digital forensics arena, it does little at this point to clarify direction.

We don't have a substantial theoretical framework beyond the information physics described earlier for identifying all of the meaningful traces and their relationships. In practice, most of today's examiners use the available tools based on their knowledge of the legal situation and how computers work to search for relevant traces and relate those traces to events. Like treating everything as a nail when the only tool available is a hammer, this approach limits what the examiner can accomplish and which traces are evaluated for which relationships.

Synergistic relationships exist between different elements of T and E so that basic properties such as independence and transitivity do not necessarily apply. Information physics needs to be developed further in order to generate the mathematical structures required to evaluate this model more meaningfully. Even if some clean mathematical formulation were in place, the legal system depends on humans to make judgments, and each case is different from almost all other cases at some level of detail. Any metrics we place on the weightings of different relationships will only ultimately be as accurate as the variance in human decision-making related to cases, and care should be taken to assure that precision does not exceed accuracy in evaluation of related issues.

There is no uniform framework for evaluation, and because of the oppositional nature of the legal process, there are always at least two parties, and often more, who act in a parallel manner and control information. A model for how to deal with these issues might stem from game theory in the form of N-player partially repeated oppositional games with limited information. Even this category of game is not well understood, and the number of strategies that may

be considered is not yet known or well understood. The limits on exploration of the total space essentially forces parties to make decisions about strategic pursuit of examination based on available information. Deception plays a substantial part in legal strategies, and to the extent that deception is in play and limited resources are available, selecting what processes to perform and when is problematic. For example, while parties don't necessarily directly lie in legal matters, it is common to withhold details of what has been done from a standpoint of digital forensics until such time as the schedule mandates presentations of reports and notices of witnesses. The presentation of large volumes of evidence within which small amounts of important traces are included is often used, and from a legal perspective, the presence of even these small traces within larger collections of evidence constitute notice of everything they may imply. For example, a trace indicative of a computer not yet identified with respect to a matter may lead to additional discovery, but if the trace or its meaning is missed, the party disclosing the evidence is not responsible for the failure to detect the trace by the other side.

Reliability figures and error rates associated with procedures are essentially non-existent in most procedures today. While digital computers have well known and widely published hardware reliability characteristics; operating systems, libraries, and applications, with few exceptions, do not have widely studied and identified characteristics of this sort. Few processes are published in peer reviewed articles, and those that are, are rarely peer reviewed or studied to the extent that tools and procedures are studied in other scientific fields. Given the large sets of possible FSM executions, it is difficult to believe that a high level of coverage will be attained by current testing methodologies. With low coverage, only statistical arguments are left, and these are problematic because the properties of errors in digital systems tend to be discontinuous and pattern specific, whereas classical statistics makes underlying assumptions about continuity and spatial distribution. For that reason, standard statistical analysis methods will not likely overcome the nature of off-by-one errors and other similar discontinuity errors that digital systems tend to have.

An apparent approach to addressing many of these challenges for limited utility is to create fault models for examination of digital forensic evidence and to use those fault models to make various kinds of assertions about results. For example, one approach to resolving reliability issues is the use of redundancy for analysis.[114] [115] Once detection is completed, a different method using different techniques and tools may be applied to generate the same result. If the independent check produces an identical result, the result is then portrayed as reliably reproduced by a second method. The opponent can independently challenge and verify or try to refute these results as well. But inherent in any such approach is the use of a subset of the model. Indeed, because of the size of the components of the model and the complexity of compositions of those components, subsets of this model will likely have to be used when specific answers to specific questions are desired.

Another approach that seems to be emerging is the creation of standard sets of examination methods and tools that can be used repeatedly in many cases and reviewed for reliability in more depth and over a larger sample set.[116] The notion of creating an increasing number of standard trace detectors and running them for efficient gathering of traces that can then be compiled into event chains to support different hypotheses about the case, appears to be the logical next step in this process.

The model and information physics

In addition to the results about this model on its own, it applies to the digital world, and thus all of the results of information physics in the digital world apply. Indeed information physics underlies the notion of internal consistency (C) in that any inconsistency must be traceable at some level to a basis in information physics or a mathematical logic that is compatible with information physics. Thus information physics is the firm basis for all internal notions of consistency and inconsistency.

---

114 T. Stallard and K. Levitt, "Automated Analysis for Digital Forensic Science: Semantic Integrity Checking", ACSAC-2003

115 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

116 "The Computer Forensics Tool Testing (CFTT) Project", Available at http://www.cftt.nist.gov/

External consistency (D) combines information physics with events. In order to make decisions within this model with regard to external consistency, there must be some way to translate the events into a form that is compatible with information physics or a compatible mathematical logic system.

It is noteworthy that information physics does not force all answers to be true or false with respect to consistency. In particular, in many cases, it only indicates partial conditions, such as that specific things cannot be true. As an example, it asserts that traces are partially ordered in that A<B and B<A cannot both be true. But it leaves the possibility of A≈B. This means, for example, that if an event asserts that A<B and traces indicate that B<A, they are inconsistent; but if the trace only indicates that A≈B, this does not confirm or refute the event A<B, it only fails to reject this event ($C_{A<B}$=0, $D_{A<B}$=0). The trace in this case is not determinant as to A<B. While the finite digital space may dictate that all problems are answerable, the answer is not limited to Yes or No with respect to any given proposition, and the size of the space of traces makes getting to all of the possible answers infeasible.

Translating words in events into testable statements

As identified above, in order to apply the model, events must be translated into meaningful statements that are compatible with the model. Information physics may then be applied to test consistency. There are some fundamental limitations on the sorts of statements that can be related to DFE in this way. In particular, anything that cannot be expressed as a finite statement in the language of the model cannot be addressed by the model, and the model alone cannot determine consistency or inconsistency. If the approach of this model is to be applied, there must be some way to turn events into statements in the language of the model.

This translation challenge is, for now, a uniquely human problem. A single event might be translated into many statements in the language of the model. As a simple example, an event such as "(e_a) The “HELO” protocols on some of the emails provided “identities” of the sending computers that do not match the IP addresses of the sending computers" might produce an arbitrary number of statements in the language used to evaluate the model. Here are

some example statements, in informal terms, that might be produced, some of which may not be true:

1. The "HELO" protocol from RFC 821 is consistent with the traces of the emails in the context of the software found in T.
2. No protocol other than the "HELO" protocol from RFC 821 is consistent with the traces of the emails in the context of the software found in T.
3. The available T is consistent with being the entirety of the traces potentially available in this matter.
4. The RFC 821 "HELO" protocol identifies sending computers.
5. T includes information on the relationship between IP addresses and computer identities that is inconsistent with the identities of sending computers indicated by T.
6. Sending computers have IP addresses that are properly represented in T.
7. ...

The list may continue for quite some length and ultimately involve much more precise statements about the presence of specific bit patterns within traces. For example, statement 1 at this level of granularity does not identify any specific traces, and in order to be tested, the statement would have to indicate specific criteria for traces of import; the elements of the HELO protocol and their execution in the software found would have to be identified from T and shown to behave in particular ways; RFC 821 would have to be shown to be the relevant basis for comparison; and the basis for determining consistency, such as the specific traces that would or would not appear, if this were or were not the case, would have to be identified. Each of these things and other similar things might have to be determined by examination of T, and such examination would no doubt involve some set of assumptions that would have to be justified.

But before going too far, it might be noticed that; statement 2 may be infeasible to determine because there is no limit to the number of possible protocols and because identifying all of the software present and how it executes in every possible situation is infeasible;

statement 3 is very hard to discern because of the size of T and the computational complexity of doing such an analysis on all possible traces; statement 4 is not accurate as to the facts based on the wording of RFC 821; and so forth. Information physics plays directly into the translation process as well. As we drive toward more precise statements, we move further from the events and risk translation failures and the potential for the presence of many different translations that involve different traces. Depending on the standard of proof, a single inconsistency may be adequate to win, or a host of them may not be enough to win.

Understanding the model in terms of diplomatics

The model of examination becomes much more useful and sensible when put in the context of diplomatics. The underlying notion of consistency is essentially the same in the model of digital forensics as it is in the relevant portion of diplomatics. While diplomatics uses the general notion of defined procedures producing records that can be verified for trustworthiness through examination of documentary form and archival bond, the model speaks of type C and D consistency, which are quite close to the same thing. Table 4.1 shows the closeness of the link between diplomatics and trace consistency as a model for examination.

Diplomatics concept	*The examination model*
Acts: exercises of will intended to produce effects.	
Transactions (acts) occur in writing, resulting in records.	The processes of FSMs produce traces, including probative records, dispositive records, and a wide range of nonlegal records.
Probative records (legal records as evidence of the act that resulted in the records)	These are often transacted in computers producing traces in the form of explicit records.
Dispositive records (legal records that put the act into being)	These are often transacted in computers producing traces in the form of explicit records.

Diplomatics concept	*The examination model*
Nonlegal: supporting records	These are typically contained in readily available or organized direct traces produced by acts. They are circumstantial but not adequate to evidence an act.
Nonlegal: narrative records	These are typically contained in readily available or organized direct traces produced by acts. They are circumstantial but not adequate to evidence an act.
Nonlegal: instructive records	These are often present and provide information useful in testing type C consistency.
Nonlegal: enabling records	
Nonlegal: enabling: performance of a mechanism	These are the mechanisms that form the executing FSMs in the form of software or firmware. As such, they should be consistent with the traces produced. Type D consistency is often testable against these records with traces in context of events to confirm or refute claims.
Nonlegal: enabling: execute business transaction	These are also mechanisms that form the executing FSMs and should be consistent with the traces produced for type C and D consistency testing.
Nonlegal: enabling: conduct experiments	Depending on specifics, these too should be testable for type C and D consistency with traces and records.

Diplomatics concept	*The examination model*
Nonlegal: enabling: analytical or observational data	Depending on specifics, these too should be testable for type C and D consistency with traces and records.
Persons: FSMs acting on behalf of persons are capable of carrying out acts.	
Persons: author	This is a person but they may vest the motions of the act into motions taken by others (e.g., the originator of a Web access).
Persons: writer	This too is a person, but may vest the motions of the act into automated mechanisms (e.g., the software)
Persons: addressee	This is often identified in traces as a “user identity”, which is often related to a human person through traces and events. Claims are often examined with respect to these traces and related records for consistency.
Persons: creator	Traces often provide an identity associated with a creator through metadata (typically the “owner”). Type C and D consistency can often be tested based on related traces and events for increased certainty.
Persons: originator	This is purely determined by context and the examination process must take into account events as well as traces in order to identify consistencies with claims.

Diplomatics concept	*The examination model*
Procedure: a formal sequence of steps by which a transaction is carried out.	
The procedure governing the act	
Petition to authority	Records or traces of such requests are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Intercession or recommendation	Records or traces of such requests are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Intervention or permission	Records or traces of such requests are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Command to create the record	Records or traces of such requests are often present in substantially redundant form. They may be measured for trace consistency in many ways.
The procedure governing documentation of the act	
Creation of the draft	Records or traces of such acts are often present in substantially redundant form. They may be measured for trace consistency in many ways.

Diplomatics concept	*The examination model*
Preparation of fair copy	Records or traces of such acts are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Registration	Records or traces of such acts are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Validation	Records or traces of such acts are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Tax computation	Records or traces of such acts are often present in substantially redundant form. They may be measured for trace consistency in many ways.
Delivery of record	Records or traces of such acts are typically kept with additional traces in substantially redundant form. They may be measured for trace consistency in many ways.
Documentary form (intrinsic and extrinsic elements)	
Extrinsic elements	
Medium (physical carrier of the record studied in terms of the material, manner of preparation, watermarks, shape, size, edging, rulings, etc.)	This is outside of the scope of DFE examination as it goes to purely physical issues. Indeed DFE is essentially independent of physical realization.

Diplomatics concept	*The examination model*
Script (i.e., layout, paragraphing, punctuation, abbreviations, and initialisms)	These are generally dictated by the specific FSMs involved, and such mechanisms are usually very consistent in the traces and records they produce. Type C and D consistency checks are straight forward for many such mechanisms.
Language (i.e., style, formulas, and tenor of discourse),	These approaches tend to be problematic as will be discussed in later chapters.
Special signs (i.e., symbols identifying persons involved with the record, like logos, heraldic markings, mottos, stamps, or drawings which are considered key to provenance)	These are commonly present as dictated by the specific FSMs involved, and such mechanisms are usually very consistent in the traces and records they produce. Type C and D consistency checks are straight forward for most such mechanisms and metadata is also often present for such checks.
Seals (examined for material, size, shape, typography, legend, and affixation method as indicators of origin and authority of the record)	These are often generated using cryptographic methods designed to be tested for verification. As such, they are ideal for type C and D consistency checks.
Annotations	

Diplomatics concept	*The examination model*
Annotations: at conclusion (e.g., the annotation in a register or book, with relevant page and date, records the identity of persons issuing the record, records of acts referred to in the record like an oath of office, etc.)	Many automated systems add annotations such as cryptographic checksums, and underlying mechanisms often include CRC codes and similar things to assure detection of alteration. They are readily testable using type C and D consistency checks in many cases.
Annotations: in use (e.g., mention of the decision made or further actions to be carried out, dates of hearings or readings, markings like “urgent”, etc.)	Such annotations are often present in computer systems and reflected in the creation of new records when affixed and sent through space. These create additional records.
Annotations: in record keeping (e.g., a registry number, classification markings, Dunns numbers, metadata, etc.)	Most digital systems maintain various metadata associated with records and their archives and these are often testable for type C and D consistency.
Intrinsic elements	
Protocol: (e.g., it's place, time, date, subject, persons participating, etc.);	For digital documents containing such content, such as structured messages, documents of identified formats, etc. test for type C and D consistency are straight forward in most cases.
Text: containing the action or message and its motivation, circumstances, or conditions;	These are outside of the realm of trace consistency and go to content and meaning of records.

Diplomatics concept	*The examination model*
Eschatocol (validation): including the means used to validate the record, the signature of the author, witnesses, and countersigners	In many cases digital systems use digital signatures that are designed to be validated through automated mechanisms and can be tested for type C and D consistency.
Other formats and consistencies	Essentially all type C consistencies fall into this category, depending on the particulars of the specific formats and FSMs that produce them.
Nature	The nature is potentially subject to type C and D consistency checks depending on specifics.
Provenance	Provenance information associated with digital records are susceptible to type C and D consistency checks.
Archival bond	
Originary	Type D and C consistency checks may be revealing with respect to the link between asserted and found indicators of the source, content, and context.
Necessary	Asserted records may not be present in the archives as asserted, and type D consistency checks are typically revealing of such discrepancies.

Diplomatics concept	*The examination model*
Determined	To the extent that the archives are properly operating, such uniqueness may exist, but at the level of DFE, it may not be demonstrable. Such issues as file system inconsistencies could cause this property to be unverifiable or untrue.
Archives consistency	Archives consistency is also at issue in digital systems. For various intentional and accidental reasons, digital archives may have type C consistency problems that put their integrity into question. Because the trustworthiness of the archives may cause all records to become suspect, this is an important area to consider.
Status of transmission	
Original record based on primitiveness, completeness, and effectiveness.	
Original: Primitiveness	This is physical in nature and no actual original records are available in digital form. They are typically treated as original for the purposes of the court if they are accurate in every sense as to the bits represented.

Diplomatics concept	*The examination model*
Original: Completeness	Inconsistencies are often revealing with respect to completeness, and complete records and their associated archives are rarely provided in civil matters. Nonetheless, consistency checks are for more likely to be revealing if complete records and archives are made available.
Original: Effectiveness	This is beyond the scope of digital evidence examination in the technical sense.
Draft	Drafts are often available and trace consistency can often be used to detect such things as ordering of drafts in time, who contributed what to which drafts, and a great deal of related data, particularly in formats and with mechanisms used to generate documents. These should be testable as consistent with the events asserted as well.
Copy	
Copy in the form of the original	In the physical sense, this is never really available in DFE, but at the level of digital evidence, bit sequences can be reproduced in such copies and can be verified as such.

Diplomatics concept	*The examination model*
Imitative copy	Imitative copies can often be tested against originals in many cases, because of presentation issues, this is problematic. The model does not provide specifics for this approach and similarity analysis discussed later in the book might be of interest for this issue.
Simple copy	This is similar to the issues for imitative copies.
Trustworthiness is measured retrospectively for digital forensics in terms of reliability, authenticity, accuracy, and authentication	
Reliability: the record as a true statement of fact	DFE examination does not speak to the truth of records.
Reliability: completeness	DFE examination can speak to this issue to the extent that the required elements can be verified in the digital form of the record. This is a type D consistency check.
Reliability: controls exercised in its creation	To the extent that there are nonlegal records related to the processes of creation, these can be checked for type C and D consistency.
Authenticity: a record has not been tampered with or corrupted	Such a record should produce no type C or D inconsistencies above base rates for loosely related phenomena.

Diplomatics concept	*The examination model*
Authenticity: preserves identity	Type C and D trace consistency should be demonstrable for authentic records. Metadata is likely to be available to verify against content, dates, and internal and external data should match to the extent that they are of similar granularity and precision and the identified FSM produces such consistent records in normal operation.
Authenticity: maintains integrity	If integrity mechanisms are in place in the form of redundant records or cryptographic mechanisms, such records should produce no type C or D inconsistencies. Other related redundant traces should also be consistent to demonstrate integrity and system operation may be tested for indicators of corruption or inconsistency in traces.
Accuracy	
Truthfulness	Truthfulness in the absolute sense is beyond the scope of DFE examination, but to the extent that there are inconsistencies, the inconsistent traces and/or events cannot all be true. Which is true is not determinable by DFE examination.

Diplomatics concept	*The examination model*
Exactness	All traces are exact in the sense that there are two and only two values for any given bit.
Precision	Precision is limited by digital physics. To the extent that precision varies between different records, such as time stamps, this can be problematic for consistency checks, which cannot be precise based on imprecise or differing precision inputs. This is addressed throughout the rest of the book.
Completeness	Completeness can only be measured against a standard, and thus type D consistency checks are appropriate. To the extent that traces indicate the presence or absence of records in the context of claims of mechanisms, type C inconsistency may be shown.
Authentication	
Declaration of authenticity	DFE examination does not speak to declarations other than to identify potential type D inconsistencies of a peripheral nature..
An element added to the record after its completion	To the extent that such elements are trusted tags or cryptographic mechanisms, examination may reveal type C inconsistencies between traces of records and added elements.

Table 4.1 - The model in the context of Diplomatics

Questions

1. After reviewing the dissertations and papers of Gladyshev, Carrier, and Kwan, comment on the advantages and disadvantages of their approaches compared to the approach identified in the model used in this book. What are their advantages and disadvantages? Has this book properly characterized the other approaches? Is the approach of this book better? If so in what ways? If worse, in what ways?
2. The legal context is, at best, poorly defined in the present characterization. How is the examiner going to deal with this lack of clear definition?
3. The notion of events (E) seems to be a bucket for depositing anything that is not a trace but that has to be related to a trace. Is this too simplistic? If so, how can it be better broken down in order to bring insight into the examination process and the notion of D consistency?
4. Given the size of T, there is no hope of exhausting the set of traces for any case that is likely to be encountered. If the space of T is not exhausted, how can it ever be claimed that the examination was complete? And if the examination is not complete, how can we ever be certain that the opposition won't find a trace that demonstrates that all of the results of examination are in fact wrong?
5. Given the size of T, the size of all possible C may be astonishingly large, and for any non-trivial matter, will be beyond any hope of thorough exploration. Is there a way to explore only select portions of this space rather than exploring the whole space? Is the notion of classes within the overall state, such as the approaches of other authors, a viable path to this? If so, what are the problems with this approach and what are the benefits?
6. Given the forensic procedures (P) that are potentially available and the enormous apparent mismatch between these procedures as they exist today and the model as defined, is there any hope of current P being useful in getting at the issues identified in this model?

7. The resource problem was apparently a key issue in Kwan, and the model used by Kwan, et. al. provides a possible path to getting at those issues, even if in a simple way. How can the present model be leveraged realistically to get at the resource issues in real cases?
8. The schedule appears to make things particularly brutal with respect to this model because the elements of the model change with time. How is any sort of optimization going to be done in when the elements of the model can change?
9. Is there a game theoretic aspect to the schedule? If so, how might this be brought out with additional treatment? What kind of game would it be? If not, how else might we deal with these issues?
10. How does the present model interact with information physics to help and hurt in the job of the DFE examiner, or is this whole theoretical thing a complete waste of time?
11. If the translation between the legal world and the theoretical world is so poorly understood that it cannot be precisely characterized, won't the whole notion of such a model ultimately be a fools errand?
12. The underlying claim of this methodology is that analytical frameworks integrated with physics ultimately produce a way to create calculation methods that have a sound scientific basis. Does this argument and approach make sense to you? If not, why not? If so why?
13. Given that an opponent may ultimately call out the inadequacies of your examination based on the methodology you use and how you apply it, how will you defend your other methodology against the claims made under this one? Is your methodology compatible with this one? It is incompatible? How will you answer these sorts of questions in court?
14. Given the closeness of diplomatics and trace consistency as an approach to DFE examination, how would you argue against the use of trace consistency as a valid basis for use in legal matters.

5 Analysis

The requirements for the use of scientific evidence through expert opinion in the United States and throughout the world are based on principles and specific rulings that dictate, in essence, that the evidence be (1) beyond the normal knowledge of non-experts, (2) based on a scientific methodology that is testable, (3) characterized in specific terms with regard to reliability and rates of error, (4) that the tools used be properly tested and calibrated, and (5) that the scientific methodology is properly applied by the expert as demonstrated by the information provided by the expert.[117] [118] [119] [120] This approach to meeting these criteria for digital forensic evidence begins with an analytical approach described here.

Starting with a bag-of-bits

The overall approach is based on the notion that, without redundancy, digital forensic evidence is really little more than a "bag of bits". Redundancy is inherent in human and current computer language, it is fundamental to the notion of syntax and the ability to differentiate legitimate from illegitimate syntax, and without redundancy, reliability cannot be assured, because with no redundancy, alteration of even a single bit anywhere could and would completely change the semantics of the entire digital universe. We have redundancy in digital systems, lots of it.

Redundancy in the bag-of-bits

Digital system hardware uses large collections of atoms and molecules to store each bit; instruction sets and memory pointers of processors have unused instruction codes and values that cause exceptions; software may use multiply linked lists, stack guards, input checking, and many other methods at each of many levels; and commonly used human linguistic constructs that have redundancy are applied in variable names and protocol sequences. When computers store content in files, most file systems track

117 The U.S. Federal Rules of Evidence.

118 Daubert v. Merrell Dow Pharmaceuticals, Inc., 509 U.S. 579, 125 L. Ed. 2d 469, 113 S. Ct. 2786 (1993).

119 Frye v. United States, 293 F 1013 D.C. Cir, 1923.

120 Reference Manual on Scientific Evidence - Second Edition - Federal Judicial Center, available at http://air.fjc.gov/public/fjcweb.nsf/pages/16

access, write, and creation dates and times, system logs produce audit information related to the execution of the programs run, files often have date and time indicators within records or other data they store, and the sequence of writes of different parts of different files leaves traces in the structures of links between allocated areas within the file system.[121] [122]

When computer networks are used, there are often, without limit, (1) time stamps placed in records by parties handling content in motion, (2) audit records in intermediate systems, (3) records from programs used to look up network addresses, (4) flow records associated with network traffic, (5) performance impacts of flows and activities on others, and (6) differences in times taken by interacting processes.

In short, there are many traces of activities that take place within the stored information in computer systems and networks. While it may be trivial to generate a bag-of-bits that is internally consistent with a set of claims in a legal matter, the creation of a complete and consistent set of all of the redundant traces from all interrelated systems and mechanisms is quite a different matter.

Moving from the bag-of-bits to a meaningful context

The challenge to the DFE examiner when facing a bag of bits is to perform various sorts of analyses that exploit inherent redundancy in the traces to find and analyze traces that are revealing with respect to the legal matter at hand. The typical analysis starts with an initial set of conclusions drawn based on an initial examination that indicates, for example, that a trace provided is from a file of a particular sort, based on name, format, statistics, metrics, classifiers, the lack of error messages from particular tools, the output of a tool, and/or other similar indicators.[123] This analysis

121 Svein Yngvar Willassen, "Timestamp Evidence Correlation", Presentation at IFIP WG 11.9 International Conference on Digital Forensics, January, 2008.

122 Svein Yngvar Willassen, "Hypothesis-based investigation of digital timestamps", chapter in Advances in Digital Forensics IV, Ray and Shenoi ed., Springer, ISBN# 978-0-387-84926-3, 2008.

123 V. Roussev and S. Garfinkel, "File Fragment Classification - The Case for Specialized Approaches", IEEE SADFE Workshop, May 21, 2009. [This paper summarizes file classification approaches and provides a case for one particular approach]

exploits the redundancy inherent in the bag of bits to allow the examiner to identify useful traces based on structure that permits further analysis. In many cases, the evidence is provided along with a set of asserted events, such as claimed facts or statements made by parties to the legal proceeding in some documented form. These events are usually closely linked to the subject matter of the legal proceeding, and as a result, the examiner is tasked with comparing these events to the traces to determine whether they are consistent or not. Consistency tends to lend weight to the accuracy of the asserted events, while inconsistency tends to refute the asserted events.

Testing and fault models as an approach

Digital system testing has improved the quality of digital systems by the development of methods that allow the reliability of digital systems to be systematically examined. Improvements in these methods allow systems to be systematically measured against models. Models are based on the underlying physical mechanisms thought to cause these faults. Observed phenomena ultimately get traced back to physical mechanisms based on the ability to repeatably detect them and perform experiments that generate those faults consistently under observed conditions.

Early digital systems testing work is well summarized in[124] which, in section IV, discusses "Automatic test generation of component failure detection and diagnostic tests". These tests are based on the underlying concepts of Moore[125] who describes the differentiation of different sequential automata. In essence, a fault can be characterized as something that changes the underlying finite state automata from a desired automaton (referred to in the testing literature as a "golden unit") to a different automaton. A test is characterized as a procedure which allows the differentiation of the golden unit from other automata. A methodology that generates tests for classes of faults generates sequences of inputs and/or states and/or conditions that allows the tester to differentiate any of

124 Melvin A. Breuer, "General Survey of Design Automation of Digital Computers", #1710 Proceeding of the IEEE, December, 1966.

125 E. F. Moore, “Gedanken experiments on sequential machines,” Automata Studies. Princeton, N. J.: Princeton University Press, 1956, pp. 129-153.

a class of golden units from other automata based on the presence or absence of members of the class of faults.

The "coverage" of a test is defined as the number of faults within the fault model that the test detects with regard to any particular automaton divided by the total number of faults feasible for the automaton within the constraints of the fault model. A test with coverage of 1 is called a "complete" test in that it covers all of the faults feasible for the automaton within the fault model. The complexity of generating and performing complete tests for various kinds of faults in various kinds of automata has been analyzed, and for even relatively simple "stuck-at" faults in common classes of sequential machines, such as those that are used to perform digital forensic analysis, the complexity of generating and performing complete tests is too high for practical purposes. However, there are many techniques for slightly altering automata for testability that make complete tests for many fault types feasible and, in most cases, relatively straight forward.[126] The field of built-in self-test[127] has been built up based on these approaches, and the techniques are now widespread in systems and commonly applied for improved reliability and early detection of faults.

An overall fault model for digital forensics has been proposed and discussed.[128] "This model assumes that digital forensic evidence is identified, collected, transported, stored, analyzed, interpreted, reconstructed, presented, and destroyed through a set of processes. Challenges to this evidence come through challenges to the elements of this process. Faults consist of intentional or accidental making or missing of content, contextual information, the meaning of content, process elements, relationships, ordering, timing, location, corroborating content, consistencies, and inconsistencies. Not all faults produce failures, but some do. While it may be possible to challenge faults, this generally does not work

126 M. Breuer, A. Friedman, "Diagnosis and Reliable Design of Digital Systems", M. A. Breuer and A. D. Friedman, Computer Science Press, 1981, Breuer, Rockville, Md.

127 The International Test Conference and many other conferences and venues consistently examine built-in self-test and a wide range of related methods. http://www.itctestweek.org/history.shtml

128 F. Cohen, "Challenges to Digital Forensic Evidence", ASP Press, 2008 ISBN#1-878109-41-3.

and is unethical if there is no corresponding failure in the process. Certain things turn faults into failures, and it is these failures that legitimately should be and can be challenged in legal matters. Failures consist of false positives and false negatives. False negatives are items that should have been found and dealt with in the process but were not, while false positives are things that should have been discarded or discredited in the process but were not." This is consistent with previous types of fault models[129] and similar to those identified in other DFE examination processes.

Feature and characteristic detection and analysis

Features and characteristics have to be detected and analyzed in order to be used by the examiner. Starting with earlier results[130] and those presented in Chapter 4, we know that for any real forensic examination, we will need to find $\mathbf{p} \subseteq P$ that allows us to identify revealing $\mathbf{c} \subseteq C$ and/or $\mathbf{d} \subseteq D$. In the general case of a bag-of-bits, we can review computational complexity associated with known procedures and, based on a set of assumptions about syntax and semantics derived from the manner in which computers are used, we can particularize these procedures and complexity measures to specific consistency and inconsistency detection problems relevant to the matter at hand.

In this context, content has characteristics, such as the file and data structures associated with the operating environment they are used in, and features, such as the specific content of files and its meaning in context. To get a sense of this, a structured file, such as a document, has:

Characteristics, like the document type and its syntax, and

Features, like the combinations of words used within it and types of spelling errors, if any.

129 “Basic Concepts and Taxonomy of Dependable and Secure Computing“ Algirdas Avizzienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr, IEEE Transactions on Dependable and Secure Computing, V1,#1, Jan-Mar 2004.

130 F. Cohen, "Two models of digital forensic analysis", IEEE/SADFE-2009, Fourth International IEEE Workshop on Systematic Approaches to Digital Forensic Engineering, In conjunction with the IEEE Security and Privacy Symposium Oakland Conference, Oakland, CA, USA, May 21, 2009.

Similarly, unstructured content, like graphical image files, have:

Characteristics, like the number of pixels contained, and

Features, like areas that look like eyes, tables, or grass.

While traces don't inherently have any of these things other than a length and the specific bits included, as assumptions are made about the traces based on events and analysis, the assumptions lead to the definition of characteristics and features. Those characteristics and features may be consistent or inconsistent with the assumptions regarding the traces and the events identified. While we will generally discuss things based on the assumption that the turning of traces into characteristics and features are valid, it is worth keeping in mind that these typing assumptions may not be valid. If they are not, the assumptions, rather than the traces and events, may be sources of consistency and inconsistency.

What is the symbol set?

One of the first questions to ask in analysis of traces is what comprises the symbol set of the traces. Most modern computer systems organize content in a hierarchical structure with bits being the lowest level entity, followed by bytes of 8 bits each, and in storage media, moving next to blocks of 512 or some other larger power of 2 in bytes. The applicable symbol set is dictated by the context of its creation and use, and thus contextual information must drive the analytical process if it is to make sense in context. In typical computer systems today, there are many different contexts and thus many different symbol sets and interpretations. For example:

- Internet traffic is organized into "octets" which are 8-bit sequences within "datagrams". But these octets are simply the structure used in descriptions, while fields within datagrams range from as little as a single bit to as large as the data structure of a datagram allows, which is 65,535 octets minus the header size.[131] Packets within networks typically include a layered set of content with each layer enclosing the next, so that, at the level of the packet there is typically a MAC address and a datagram or datagram

131 RFC791 - "Internet Protocol", at http://www.faqs.org/rfcs/rfc791.html

fragment. Within each of the MAC addresses and the datagram fragments there are additional fields of different sizes depending on the specifics of the situation.

- Data structures within memory and storage are configured based on the computer programs and hardware mechanisms that use them.
- Microsoft files, such as word documents, spreadsheets, and powerpoint presentations, use an object linking and embedding (OLE) format containing fields of various lengths in various formats, and these vary from version to version.
- Mail transfer agents use largely text-based storage, while user programs that present and handle email for users tend to use different formats and symbol sets for their representations.
- Databases use internal data formats that typically include multiple files organized into different structures using different symbol sets.

The list of different formats and symbol sets is potentially as large as the total number of FSMs that handle content.

When more than one program or FSM uses a data structure, there are complex interactions, sometimes causing errors, and sometimes with one or the other of the mechanisms using the data structures applying different things about them. Multiple symbol sets may also be simultaneously active, particularly when there is inherent redundancy in the syntax of one or more types of content that can be used by another syntax simultaneously.

Computers are, in one way of thinking, symbol processing mechanisms that can handle arbitrary symbol sets, including symbol sets with symbols of different length and with multiple meanings. Even the representation of a string of characters like a sentence can vary dramatically, with different encodings of different sizes for the characters, different ways of indicating the length and end of the string, and different hardware and software used to interpret and manipulate the string.

Unless and until the analyst makes some hypotheses about the symbol set(s) and interpretation mechanism(s) in use, it will be meaningless to try to undertake analysis. But these hypotheses do not have to be uninformed. There are many clues as to what makes a reasonable hypothesis, starting with the events, which may include information given by witnesses, or other statements or assumptions made with regard to the issues in the case. One side may have proffered traces indicating that they are an exact copy of an HTML[132] page loaded from a particular Web site at a particular time. In this case, the analyst has something to work with to start to make sense out of the bag of bits. If these assertions are true, there should be no inconsistencies indicated by the traces that would rule out the use of the identified traces within a Web page. Without an initial hypothesis, the examiner can form hypotheses by trying to "type" the trace, perhaps by using a series of hypotheses relating to different known symbol representations and testing each one to find those that are consistent or inconsistent with the available traces and going from there.

Trace typing

Traces are commonly "typed" before being further analyzed. The underlying syntax of the media typically leads the analyst to examine portions of the traces as groups, such as blocks or subsequences of other sorts, and this in turn leads to identification of likely types such as file systems, files, embedded files, logs, messages, and so forth. This typing effort is fundamental to the creation of assumptions that the examiner uses for further examination of traces, and there are several common methods used to do this typing.

Typing of media is generally initiated based on sequences associated with the headers placed by the FSMs that generate them, so as to make identification and proper use easy. But for various reasons, these headers may be inconsistent with the content or otherwise misleading. Header or other meta-data examination, file names, and similar indicators of data type are almost all O(1) in complexity when applied to a particular sequence of bits. For files or embedded file systems, headers are also

132 HyperText Markup Language (HTML) is the syntax used for Web pages as defined at http://www.w3.org/

commonly used, and to type content in the storage hierarchy that is commonly used in most digital systems today, takes as much time and space as spanning the tree within the trace.

Other methods of typing include syntax analysis by (1) content examination using heuristic methods like the "JDLR" (Just Doesn't Look Right) techniques from ForensiX,[133] (2) statistical analysis such as information content measures,[134] more specific statistics purposed for differentiation of content type,[135] [136] or "learning classifiers",[137] most of which are normally linear time O(n+m) for n different types and m bits of content, or (3) the application of state machines built to parse different syntaxes, such as the use of multiple lexical analyzers, which are also usually O(m+n). Because multiple types may be simultaneously conjoined in the same sequence, even "correct" detection may not tell the whole story.

Inconsistencies within the type information are problematic in that (1) there are many possible causes, and (2) without a consistent set of types for portions of traces, the analysis is reduced to all possible interpretations of all possible traces. In most legal matters, type information is indicated by events. For example, the files provided as items of evidence may be from an individual's "Windows" system. This establishes an event that can be confirmed or refuted as to type by examination of traces using the methods identified. However, just because the type information is consistent with the events, doesn't make this the only interpretation of the traces. For example, there could be covert information such as steganographic content, the same information might have different interpretation in a different context or when interpreted by different FSMs, and the event information is often not complete and precise.

133 F. Cohen, "ForensiX", The ForensiX Just Doesn't Look Right (JDLR) mechanism is detailed in the source distribution available in http://all.net/ForensiX/Forensix.tar

134 S. Moody and R. Erbacher, "SADI – Statistical Analysis for Data type Identification", 3rd International Workshop on Systematic Approaches to Digital Forensic Engineering, 2008.

135 V. Roussev and S. Garfinkel, "File Fragment Classification - The Case for Specialized Approaches", IEEE SADFE Workshop, May 21, 2009.

136 W. Calhoun, D. Coles, "Predicting the types of file fragments". Digital Investigation v5, 2008, ppS14-S20.

137 See "International Workshop on Learning Classifier Systems (IWLCS)", an annual conference on learning classifiers and related methods.

Virtualization may be used or the system might have been bootstrapped from different media at different times, each producing multiple sets of FSMs interpreting the same traces. Thus the underlying FSMs operating are not definitively determinable, even if consistency is maintained within the context observed.

Imitative copies, regular expressions, and similar analyses

For obvious redundancy, such as imitative copies (also called exact) of identified content in the same format at defined locations, finding duplicates is relatively easy. The $_{P}\subset P$ is a simple linear time bit sequence match between traces at known locations in a random access media. If the bit sequences match, they are consistent at level 1, and if they mismatch, they are inconsistent at level -1, where the levels are the value of the relation defined for C. Complexity is linear at O(m) where m is the length of the sequence searched. A consistency measure that is more tolerant of deviations might identify the extent to which the sequence of bits match, for example, by associating +1 with all bits matching, -1 with no bits matching, and the linear range between these extremes with the prorated value of matches per bit. But this is problematic in that, in a non-continuous space, such as the space of digital values, a single bit can completely change the syntax and semantics of content, depending on the inherent redundancy of the language in use. For compressed or encrypted data, for example, a single bit change can produce a completely different interpretation.

More generally, searching for a string within a larger string is also linear at O(n+m) where n is the length of the text to be searched and m is the length of the text to search for.[138] A similar approach may be taken for the detection of contraband, where we are identifying some $_{D}\subset D$ where $_{T}\subseteq T$ matches specific defined bit sequences identified in some $_{E}\subseteq E$. If they match, they are consistent at level 1, and if they mismatch, they are inconsistent at level -1, where the levels are the value of the relation defined for D. This constitutes a substantial portion of the current digital forensic analysis effort for cases involving contraband, such as cases involving child pornography, classified information on unclassified

138 P. Weiner, "Linear pattern matching algorithm". 14th Annual IEEE Symposium on Switching and Automata Theory: 1-11. (1973).

systems, unauthorized access devices, and similar sorts of possession-driven offenses.

Considerably better results can be attained by searching the same set of traces for multiple strings or, more generally, for sets of patterns of the class that can be written as regular expressions. Regular expressions can be searched for in linear time through the use of sequential machines.[139] A wide range of similar search methods that gain faster time for repetitive searches of the same traces are also identified[140] and have been substantially improved upon over time. The use of hashing algorithms and similar methods make searching for similar strings very fast as well. The underlying assumption associated with regular expressions is that the syntax of interest can be codified in terms of a regular expression. While this is often true for some formats, like headers in electronic messages, for other formats, such as graphical image formats, this is far less effective.

Using a message-specific example, searching message headers for sequences associated with a particular header, such as the "Received:" header, is straight forward using string matching, and parsing this header according to its specification. For email, those specifications are most often specified in requests for comments (RFCs) 821[141] and 2821.[142] Searching common protocol formats used in the Internet is straight forward using regular expressions or look-ahead left-right (LALR) lexical analyzers.[143] Similarly, the email sequences within a mailbox can be separated into individual email sequences (consisting of a "From " separator, a header, and a body each), and other similar parsing operations can be carried out to create collections of different portions of the email sequences.[144]

139 P. Weiner, "Linear pattern matching algorithm". 14th Annual IEEE Symposium on Switching and Automata Theory: 1-11. (1973).

140 D. Knuth, "The Art of Computer Programming, Volume 3, Searching and Sorting", 1973, Addison-Wesley.

141 J. Postel, "Simple Mail Transfer Protocol", RFC 821, Aug, 1982. http://www.ietf.org/rfc/rfc0821.txt

142 J. Klensin, "Simple Mail Transfer Protocol", RFC 2821, Apr, 2001, Available at http://www.ietf.org/rfc/rfc2821.txt

143 D. Knuth, "The Art of Computer Programming, Volume 3, Searching and Sorting", 1973, Addison-Wesley.

144 The mbox format is specified at http://www.qmail.org/qmail-manual-html/man5/mbox.html

For messages in these formats, these processes are all linear in the length of the mailbox file. Parsing operations may yield inconsistencies and, to the extent that they do, these go to both C and D consistency issues.

It is useful, from a consistency analysis standpoint, to produce different sorts of content associated with messages as collections, referenced to the original traces. For example, from a trace that is a mailbox, parsing the trace into collections of email sequences extracted from the original trace and associated back to that trace is useful because each email constitutes a syntactic element. Within each email, there are syntactic elements such as the separator, header, and body, and within the header there are syntactic areas called headers. Each of these and their syntactic components can also be parsed so that the fields within those components are separated and identified, including being marked as to their origin back up the syntactic tree. For example, the date and time stamp within a "Received:" header from an email within a mailbox might be made part of a collection of times with a back reference to the header number within the header of the email sequence extract number within the mailbox. These back references are useful in doing subsequent consistency analysis. If each of these are linked back to the original trace and if bit-for-bit accuracy is maintained in the process, these sequences are themselves traces that can be analyzed both independently and in context to identify C and D types of consistency. If the parsing can be done with an LALR parser then the complexity is limited for parsing, and such parsers can handle parsing for all "Backus-Naur Form" (BNF) specifications,[145] such as those used in many RFCs.

In the process of parsing emails, various errors may occur in the parsing process. These errors demonstrate either a problem with the parsing procedure, internal inconsistency within the mailbox, or inconsistencies relating events to traces. For example, if the mailbox does not start with a "From " separator, this is inconsistent with the mailbox format; if lines within the mailbox file are longer than 80 bytes each, this is inconsistent with the format; and if there

145 Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005. This also references RFC 733 and 822 as source documents. Available at http://www.ietf.org/rfc/rfc4234.txt

are non-ASCII characters in the mailbox, this too is inconsistent with the mailbox format.[146] Such errors must not stop the processing of the traces, and the result is that there may be multiple ways of interpreting the available traces. This then adds to the complexity of analysis and, in general, makes it as complex as the possible interpretations of languages. Results for complexity analysis described above generally do not apply to the set of all error handling approaches to parsing of syntactic trees, even under grammars such as BNF.

For data, such as pictures or sounds stored in digital formats but representative of real-world content captured as depictions, exact matches are far more interesting. Of course the notion of parsing a picture is very different from that of parsing a data format, but some forms of parsing are used to, for example, detect lines, identify shapes, and for other similar purposes. These sorts of analysis are completely different for data of this sort than for structured data associated with expressions of language with defined syntax. Few of them have linear complexity with the number of bits in the image, and the notion of consistency is far more complex, going to the issue of what the image represents, rather than the mere presence of bits in locations. Identical copies can be detected with methods that are linear time for a fixed set of comparisons, such as the search for known images of child pornography, graphical images like icons known to exist in particular operating environments, and even inked tracers[147] placed in digital output by printers and then scanned using higher resolution imaging devices.[148] But these are the exceptions rather than the rule for such analysis.

146 The mbox format is specified at http://www.qmail.org/qmail-manual-html/man5/mbox.html

147 The term "tracer" is used in this book to indicate an intentional marking or pattern placed for the purpose of identification or attribution of devices and/or methods. It is not a commonly used term in the literature today. In the case of color laser printers, "Machine Identification Code" is sometimes used, while a Media Access Control (MAC) Address is associated with network interface cards, serial numbers for various media, and other similar tracers in other situations. The term "tagent" is used in a similar manner in other contexts.

148 D. Schoen, "Investigating Machine Identification Code Technology in Color Laser Printers", 2005, The Electronic Frontier Foundation, available at: http://www.eff.org/wp/investigating-machine-identification-code-technology-color-laser-printers

As a simple experiment, we repeatedly scanned the same piece of paper 9 times on the same flatbed scanner without delay and without moving the paper. Each resulting scan file varied in length and content from every other scan file. At the level of 16 byte chunks, the files differed in 99.96% of chunks. The first 256 bytes were identical headers in all of the scans, and only 153 other chunks matched across files, these matches distributed throughout the files and across different pairs of files. Even the same input device yields different outputs for the same source, so exact matching is clearly a problem for these sorts of inputs.

Equivalent content in different formats

For less obvious redundancy, several challenges remain. A slightly harder problem than searching for exact matches is searching for the same content in different formats or other similar inexact matches. This class of searches divides the space into equivalence classes and searches for members of those classes. Assuming the classes can be characterized in terms of syntactic elements that can be parsed by an LALR parser, such as a BNF specification, linear time results apply as above. But not all equivalence classes can be readily specified in this manner. For example, human language cannot be accurately parsed this way.

As an example of LALR parsable content, date and time stamps come in a wide range of different formats, and one of the key issues in legal matters tends to be the timing and ordering of events. Time and date stamps, even within such similar records as "Received:" headers within messages, may create challenges. These time and date indicators are generally listed in a standard format, but have time zone indicators that are optionally placed at the end of the time and date stamps indicative of offset times from universal coordinated time (UTC).[149]

Even comparing such date and time stamps for ordering requires that the data be processed into a standard format prior to determining ordering. In the process of doing this reformatting, anomalies may be detected. Format anomalies constitute inconsistencies between the traces and claims that the traces are

149 J. Postel, "Simple Mail Transfer Protocol", RFC 821, Aug, 1982. http://www.ietf.org/rfc/rfc0821.txt

reflective of identified events. Such anomalies bring out the notion of comparing the traces to the software purported to have created them. For example, if a claimed event (e_1) asserts that the same method and system was used to generate a set of "Received:" headers, but the trace (t_1) indicates that the formats of those "Received:" headers differs from header to header, then t_1 and e_1 are apparently inconsistent. But that result is not always definitive.

The complexity of detecting format differences depends on the specification of the format; however, in most cases, such as the format of a message header, the time to detect the presence of different patterns is linear in the number of patterns. This is because these patterns are described in the relevant RFCs in "Backus-Naur Form" (BNF)[150] and BNF expressions are verifiable against syntax in linear time with the length of the tested string.

Normalization

The most common approach to reconciling different formats is called "normalization". The goal of such normalization is to find a format to which related record types can be transformed, so that all sources are commensurable to the normalized format for analysis. The goal is to produce a format that allows efficient and simple analysis of relations, like ordering or matching, where there is a reason to apply such a relation in analysis.

For ordered syntactic entities, like date and time stamps, where there is a strict "<" relation and multiple hierarchical fields, selecting a common format, like "YYYY-MM-DD-HH:mm:ss.pppp..." (4-digit year, 2-digit month, 2-digit day, two digits each of hours, minutes, seconds, and fractions of seconds as available) is particularly useful, because it sorts both alphabetically and numerically to the same ordering as the ordering of time.

In storing headers from messages after they are normalized (for example by turning header lines that are continuations of previous lines into a combined single line), a trace number, followed by the message number in sequence of occurrence in the trace, followed by the line number in the header of the message, followed by the

150 Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005. This also references RFC 733 and 822 as source documents. Available at http://www.ietf.org/rfc/rfc4234.txt

data from the header, provides a convenient way to sort the headers by different fields.

In the more general case, a database may be used to associate different properties with different sets of content after normalization, and database operations such as unions and intersections can be run to perform simple types of analysis. More complex analytical processes will likely not gain advantages from the database approach, but there is convenience in the use of a database to assist with traceability back to original traces, etc.

These transformations of traces into normalized forms makes them more suitable for analysis, but without the ability to link them back to the original traces, they are problematic for forensic purposes. The normalization process should also track differences in original formats, because this can be used to find inconsistencies. For example, if events assert that the time and date records are made by the same mechanism but they are in different formats, this is a potential type D inconsistency between events and traces.

Generating characteristics and features of structured traces

For structured data, even if formats are consistent in that they don't violate BNF or other syntax specifications, content may vary and provide indicators of their origin. For example, many message headers include names of software mechanisms, version numbers, IP addresses associated with identified activities, and other similar indicators.

Formatting in fields may differ based on the software used. For example, log entries and their formats vary with the software used, configured settings, and so forth. Each of these and any number of other similar characteristics may be examined, once identified as potentially relevant traces, by searching for their presence or absence. This is typically done by using a regular expression or similar descriptive method, and executing a linear time algorithm to detect the presence or absence of a characteristic in any given trace. This then generates a set of characteristics of different traces that can be related to each other in different ways.

The time consumed for searching for multiple expressions over the same trace is the sum of the times of the searches, which in these

sorts of cases can be linear in the combined sizes of the search patterns and the traces. However, in cases when different traces are used to identify different properties, the traces may not be done together unless those traces can be extracted from a larger trace in linear time during execution.

Generating lists of message headers, bodies, and locations is a special case of generating features, and this particular special case is often useful in investigations. Messages are commonly broken into headers, which contain a variety of metadata, and bodies, that contain the information being communicated end-to-end, and quite commonly, person to person. Headers and bodies are typically treated differently because headers are generated by mechanisms designed to provide meta-data in structured formats, while bodies in human to human communications contain natural language syntax and semantics associated with human communications.

For headers, a method that has proven effective is to generate a list of all headers for each message retaining the "entry number" of the header within the message and the message number of the message within a collection of messages. This retention allows the specific traces to be identified later, which will be required for presenting evidence with regard to the traces without finding them again. Derived traces are then stored in separate files or database entries, and separated by header identity for comparison and analysis. While traces of individual messages are available within each message, by storing headers in the order of the messages in the message file (e.g., mailbox) into files or collections associated with the header types, a sort by sequence of headers is provided for header to header comparison across messages, and the time remains linear in the size of the trace. The use of files for storage of results from these sorts of activities is often required because of memory limits when traces consist of large numbers of messages being evaluated against events. This approach also allows for efficient parallel analysis by breaking the original trace into sections and then performing this process in parallel. Results can often be appended in the same order they were broken apart, with almost linear scalability.

For bodies, either automated mechanisms are used to generate the content, or humans are involved. Automated mechanisms usually

have structured syntax, while humans tend to have less structured and more error-prone syntax. People tend to be good at differentiating what they believe to be obviously automated from obviously human messages, but this may be very time consuming for large collections. As a result, automation may be fruitfully applied to try to differentiate these. For automated message content bodies, similar methods to those used for structured data may be reasonably applied. A syntax tree is developed for parsing, and the parsing carried out to seek structure. This tends to be complicated and take times that are far higher than the linear times needed for simple headers such as those described above. Generating a parser is greatly eased by the use of compiler compilers and similar language parsing tools, such as the Unix "yacc"[151] and "lex" programs.[152] Other more targeted tools are typically used for parsing XML[153] and other widely used and more specific syntax structures. These are typically LALR parsed with resulting linear complexity O(n) for trace length n.

Human syntactic elements are used when people generate content. While standard language analysis has become quite advanced in recent years, analysis of messages used in messaging systems today have grown to include specific syntactic elements used for short message service (SMS) and similar low bandwidth or hard-to-enter input and output (I/O) mechanisms, such as cellular phones and instant messaging systems. These messages tend to have syntactic elements that are abbreviations that have meaning in context and are part of very quick exchanges. Parsing them takes LOL and a linguistic database and syntax structure that is far different from standard language structures. Perhaps a macro programming language would work well in this role, but we are unaware of any definitive study in this arena.

The first step in this process is to define the goals associated with the investigation with regard to the content. If semantic meaning is

151 Johnson, Stephen C. [1975]. Yacc: Yet Another Compiler Compiler. Computing Science Technical Report No. 32, Bell Laboratories, Murray hill, New Jersey. A PDF version is available at ePaperPress.

152 Lesk, M. E. and E. Schmidt [1975]. Lex – A Lexical Analyzer Generator. Computing Science Technical Report No. 39, Bell Laboratories, Murray Hill, New Jersey. A PDFat ePaperPress.

153 XML Schema and parsing tools at: http://www.w3.org/XML/Schema

important, a linguistic expert is required, but if the goal is simpler, like to find duplicate or similar bodies or parts of bodies within multiple messages, the task can be automated. Perfect matches of complete bodies can be done in linear time and space by using a hash insert of the entire body into a hash table and match detection for hash hits. The use of a cryptographic checksum or other long output space hashing algorithm (on the order of the square of the number of entries to avoid the Birthday paradox problem) can be used to produce a list of hashes which can be sorted by hash value to produce likely matches in far less space. The hash results may be retained for further association using fixed length fields, which reduces allocation overhead for large collections. Imperfect matches are far more complicated, and many approaches have been identified for imperfect matching as part of the research into Internet-based searches in search engines. For forensic purposes, depending on the specifics of the matter at hand, line-by-line matching, word sequence matching, spelling error matching, syntax fault matching, and other similar methods may be applied.

A common thread among many of these methods is to break the content into smaller normalized chunks, like words, which we will consider syntax elements, or symbols in the symbol set. Matches between counts and frequencies of symbols are commonly used to detect similar content. Symbol pairs, triples, and more generally, n-tuples may be sought to find "similar" phrasing. This is particularly useful for finding common sequences across content chunks. Some pseudo-random generation methods may be detected by looking for sequences containing one of each of sets of different collections of symbols, such as words, in sequences. In essence, all of these techniques are of complexity O(n log n) where n is the number of symbols, for any given symbol set. But the complexity goes up as the number of different symbol sets increases. Since the total number of possible symbol sets is O(m) where m is the number of bit sequences that can be chosen for symbols, and the number of bit sequences identifiable is the size of the space of traces (T), the complexity of the general class of all such matches is too high to be practical. Similarity analysis is discussed further in the "Interpretation" chapter of this book.

Typically, breaking informally structured texts into lines, sentences, and words, or the equivalent sorts of entities for digital data, are the limits of forensic analysis unless and until further information is understood about the nature of the issue and the content at hand. These are then matched across content chunks, potentially correlated with header information, and the results used to seek consistencies and inconsistencies with purported events.

For example, if different "signatures" appear in messages sent from or texts generated by the same individual, the presence of different signatures is somewhat inconsistent with the assertion that the messages were sent from or texts created by the same mechanism. Similarly, the claim that all of a set of messages or texts came from different sources and were unaltered in processing is inconsistent with the presence of identical sequences in each of the messages or texts. Each may have a clarifying explanation that provides a consistent set of events, and this sort of rehabilitation is one of the reasons that the consistency and inconsistency approach is useful rather than absolute claims. By adding events or traces, the consistency or inconsistency may change, and there is rarely a case where it is impossible that more traces or events could not, at least potentially, change the consistency results.

Generating characteristics and features of unstructured traces

For unstructured data, such as graphical images or sound files, there are different sorts of characteristics that can be generated. The "syntax" of images includes things like the color model and pallate used, the number of colors, their values, the number of dots per inch, the image size in two dimensions, orientation, color depth, compression, and any annotations, such as embedded time and date stamps, camera identifications, and so forth. Other computer-analyzable features include the results of camera and feature settings, like if and how edge lines are anti-aliased, microprinting and fonts used, steganographic content such as printer tracers, overt and covert digital watermarking characteristics, and similar sorts of features.[154] [155] All of these unstructured data characteristics

154 Rudolf L. van Renesse, "Optical Document Security", 3rd edition, 2005, ISBN 1-5805-258-6, Artech House, Boston, London.

155 F. Meng, X. Kong, and X. You, "A New Feature-based Method for Source Camera Identification", IFIP WG 11.9, International Conference on Digital

are quantifiable in fixed time or linear time in the number of pixels in the image.

Recent results from companies like Google have provided linear time parallelizable image characteristic analysis and searches for terms like "house" or "dog", these based largely on recent developments in human cognition.[156] But these methods, while useful for generating initial identifications that can be examined in more depth, are not forensically viable today beyond that purpose. The mechanisms that drive them are not statistically characterized in terms of reliability for purpose, and the purposes of generally identifying a dog or house are not probative in forensic cases beyond what an unskilled person can see by visual examination.

For graphical images, derivative traces may be generated by analytical processes and grouped together as well. For example, a line detection algorithm may identify regions of an image and they can be grouped by different image features, such as size, color, shape, density, and so forth. Just as we can build up syntactic entities in artificial data sets, naturally sourced data, such as graphical images, can be built up from lower level components to higher level syntactic entities which can be compared for consistency. For example, shadow detection has been used to determine whether image areas are consistent in terms of apparent sources of lighting.[157] Searching for tracers associated with particular printer types and particularization to specific printers with particular time stamps[158] is an example where image data is structured after low-level traces are translated into higher level syntactic elements. In this case, the tracers are in the form of repetitive patterns of particular colors across a page. These

Forensics, 2008 appearing in "Advances in Digital Forensics IV", I. Ray and S. Shenoi, Ed.

156 Tom Dean, "Disruptive Perspectives on Biological and Machine Vision", Keynote Address at HICSS 42, Jan 5-8, 2009.

157 Hany Farid, "Digital Image Forensics", National Academy of Sciences, Annual Meeting Symposium, Legal/Forensic Evidence and Its Scientific Basis, 2006/04/05, see: http://progressive.playstream.com/nas/progressive/2006am-forensic-farid/Hany_Farid.html

158 D. Schoen, "Investigating Machine Identification Code Technology in Color Laser Printers", 2005, The Electronic Frontier Foundation, available at: http://www.eff.org/wp/investigating-machine-identification-code-technology-color-laser-printers

patterns can be correlated across the page to identify internal page consistency, and can be correlated with known samples from particular printer types for typing. Particularization is also possible if the coding for placement of tracer components is known. Each of the identified tracers can be identified as higher level traces linked back to image data at locations on the page represented by the image formatted file within the file system. For obvious sorts of known tracers, such as the presence of dots at or around locations on a picture or time stamps in headers of image files, as for known byte values, the search processes are linear with the size of the graphical images, but they may take significant amounts of time to complete because of the high data volume associated with high quality images. The Google approach to image analysis may also be used to identify features,[159] and these analytical results may be compared to events such as statements about the appearance of an object, to help guide the investigator in identifying type D inconsistencies.

Features that are not so easily analyzed include properties of an image used for human comprehension and features that can be mathematically characterized but not easily located by automation. For example:

> (1) Shadows in images may be used to show the source of lighting, and the apex of the features and their shadows can be used to determine if different light sources are involved in different parts of an image, but they are hard to detect automatically.
>
> (2) Reflections from eyeballs, silver spoons, and similar highly reflective surfaces in pictures can be mapped into images of what is reflected in them and compared to each other to find composite images. But identification and analysis of these features is quite complex and not highly automated today.[160]

159 Tom Dean, "Disruptive Perspectives on Biological and Machine Vision", Keynote Address at HICSS 42, Jan 5-8, 2009.

160 Hany Farid, "Digital Image Forensics", National Academy of Sciences, Annual Meeting Symposium, Legal/Forensic Evidence and Its Scientific Basis, April 25, 2006.

(3) Finding areas within images and converting them into maps of real-world objects takes more than linear time.

(4) Analysis of facial features and similar biometrics requires substantial analysis to find the features, even though mapping into a database of features is then relatively fast. For presumptive testing, the presence of flesh-tones has been used to detect potentially human features.[161]

(5) Tamper detection by blur estimation has also been successful.[162]

(6) Image authentication systems have been proposed for tracing images to sources, and detection of sources have been experimentally performed with limited success.[163]

The complexity of these methods can be greatly reduced if manufacturers assist in the creation of identifying transforms and tracers within their devices.

Similar set searches

Similarity-based search methods are used when the examiner is unsure of the specific form of content but believes that there are various applicable forms that might be meaningful. Examples include, without limit; word stemming, in which a search term is replaced with a "stem" that will work for different word forms (e.g., theoretical → theor[a-z]*); phonic searching in which phonemes are substituted for spellings so that other similar sounding names or words will be found (e.g., "gh" as in enough, "o" as in women, and "ti" as in fiction will be found if they appear together when searching for things that sound like "fish" (e.g., ghoti); synonym searching, in which synonyms are replaced for words (e.g., a search for "find"

161 Abhishek Choudhury, Marcus Rogers, Blair Gillam, "A Novel Skin Tone Detection Algorithm for Contraband Image Analysis", 3rd International Workshop on Systematic Approaches to Digital Forensic Engineering.

162 Dun-Yu Hsiao, Soo-Chang Pei, "Detecting Digital Tampering by Blur Estimation", Proceedings of the First International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE'05), 2005.

163 I-Chuan Chang Bor-Wen Hsu and Chi Sung Laih, "A DCT Quantization-Based Image Authentication System for Digital Forensics", Proceedings of the First International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE'05), 2005.

would also find “search”).[164] Other substitutions are also used such as searches including common spelling errors (e.g., speling), widely used substitutions (e.g., effect → affect), and symbolic representations (e.g., :), 1ike, 0n3d). Most such methods can be implemented using linear time algorithms, just like regular expression searches, except that they use regular expression-like synonym lists or other similar mechanisms.

An extension of these methods is to apply such sequencing to the partially ordered set of word sequences that form phrases, sentences, paragraphs, and documents. For example, sentence structures can be formalized in to sequences of class sets (e.g., Dogs walk in parks → (noun, verb, adjective, noun)) and “concept” searches undertaken (e.g., ([animal] .. [acts] .. [place]) where [animal] is words with stems of {dog, cat, …}, acts is any verb, and place is all synonyms for any of {park, house, car}) to find all phrases or sentences of this form. If all of these are finite sets and enumerated, the search time is again linear in the size of the trace. These sorts of methods may be efficiently implemented with lexical analyzers and compiler compilers or similar tools.[165] [166]

Analysis of indicators and identifiers

Many different sorts of indicators and identifiers may be present in records. An example of an identifier that is commonly present in message traffic is a "Message-ID". Similarly, many document formatters have document numbers or other similar identifiers generated at creation time. These identifiers (IDs) are typically placed by software mechanisms as part of the creation or delivery process. They often contain sequences indicative of the software placing the ID, a configurable but often defaulted string, a separators, and numerical or other similar incrementing counters.

---

164 E. Casey, Holley, James O.; Luehr, Paulk H.; Smith, Jessica Reust; Schwerha, Joseph J. IV, “Handbook of Digital Forensics and Investigation - Chapter 3 - Electronic Discovery”, Academic Press, 2010.

165 Johnson, Stephen C. [1975]. Yacc: Yet Another Compiler Compiler. Computing Science Technical Report No. 32, Bell Laboratories, Murray hill, New Jersey. A PDF version is available at ePaperPress.

166 Lesk, M. E. and E. Schmidt [1975]. Lex – A Lexical Analyzer Generator. Computing Science Technical Report No. 39, Bell Laboratories, Murray Hill, New Jersey. A PDFat ePaperPress.

For example, JSW-TC-00005a might be placed in a twitter message to indicate Joe's Software (JS), Twitter client (TC), and the sequence number 5a (presumably preceded by 59 and followed by 5b). The same or a different indicator may or may not be used for tracking purposes in delivery and reception headers, audit trails, and/or replies.

Such indicators are often associable to the type of software and system and, in some cases, to the particular system and software, and localizable to a time frame and placement in a sequence of events.

This sequence-related information is then comparable to time-related information, and consistency may be checked for confirming or refuting hypotheses about what took place, in what order, when, and through what mechanism. In order to undertake such analysis, a translation between the format of the indicator and the type or particular system must be identified. Testing, documentation, and/or code inspection may be used to determine the ordering characteristics, if any, of the indicator. Once, and assuming, ordering characteristics are identified, ordering may be done by a sort, where the complexity is $O(n \cdot \log(n) \cdot c)$, c is the complexity of the comparison method used to determine the ordering relationship, and n is the number of items to be ordered (sorted). In most systems, the comparison is relatively simple, such as an incremental integer, while other authors choose to use the time-stamps of systems directly in their identifiers, making time correlation far easier. Some authors choose to apply pseudo-random number generators or other similar methods for sequencing, and in such cases, the complexity of determining the ordering, especially when only parts of the whole sequence are present, may be substantially higher.

Matching IDs to time within defined bounds is often feasible if there are anchor events. Inconsistencies may be detected by finding IDs not properly aligned with time stamps, or by sequencing that is out of order relative to times.

Deception and falsification of most such identifiers is simple, involving only the placement of a string of a particular format in an appropriate location within the content. But it may be far harder to

sustain such a deception in the presence of redundant records or when coding schemes are used. Cryptographic methods may make forgeries very hard, and when examiners analyze redundant records, like log files, network events, and ordering of related content, forgery may also be very difficult to do without detected inconsistencies.

Consistency analysis of characteristics and features

Once characteristics and features are identified, extracted, and analyzed in preliminary ways, whether for structured or unstructured data, the analysis focuses on identifying consistencies and inconsistencies of those characteristics and features in the general sense, and in many cases, the more specific correlation of identical, similar, and related types of features and characteristics within and between content and sources. There are many different approaches that may be used, and each has the potential to point out different consistencies and inconsistencies.

Ordering assumptions and detection of out of order entries

Time is a physical reality that impacts almost every case because most legal issues involve causality in one form or another. Such simple rules as "A caused B implies that A precedes B in time" are very powerful when there is a great deal of data related to times and events. Time is sometimes complicated in digital forensic analysis because the time bases that create time stamps within different systems and mechanisms may be of different formats, be from different time zones, have different clock skews from accurate times as defined by standards bodies, and so forth. In addition, time sequences within computers may be complicated by prior state, loads, external and internal states, inputs, processing, user intervention, and alteration of traces between their origin and delivery to the examiner.

Different sorts of similar traces are often revealing because they allow easy comparison of redundant content. For example, sequential storage files are usually generated in append operations so that the time ordering of entries normally corresponds with the sequence of entries in the file. When traces such as "Received:" headers, message separators, or meta-data have different sequences than the sequence of arrivals or have different internal

sequences than each other, this is an inconsistency. If the file is asserted as original writing[167] and not hearsay, this sort of inconsistency tends to refute that claim because the traces are inconsistent with the asserted events. The original writing problem stems from the normally sequential nature of records in archives.

Time stamps from the same computer are normally in sequence. Significant changes in ordering not explainable by the presence of other mechanisms are inconsistent with the assumption that time moves forward and processing is sequential. The hearsay problem stems from the problem that the file has to be original writing (e.g., a normal business record) or meet some other hearsay exception.

To the extent that regular expressions may be used to describe the formats of times contained in records, linear time in the length of the trace is adequate to extract the records in the order of placement. Linear time is required to detect the presence of and identify out-of-order records in the archives as well, since each must follow the previous under a strictly local comparison. Differences between ordering of sequence numbers and other similarly ordered traces can similarly be detected in this way. While detection of out-of-order entries is simple, the number of possible original orderings is, in general, the set of all graphs with nodes corresponding to time stamps. This means that rehabilitating the evidence by identifying specific mechanisms that may have caused ordering failures may be far more complicated.

Still, care must be taken in making the assertion of out-of-order records. Normal mechanisms, such as the file locks used to force sequential output in files, may cause output from parallel processes to be entered into a file in a different order than the order in which they arrived and the time stamps were placed within those entries. The inherently problematic nature of getting accurate times with similar format and precision across computers and mechanisms may also limit the precision with which ordering may be assured. A key concept is that larger variations in time tend to be harder to explain and thus to rehabilitate. In time analysis, for cases where ordering variations are critical at high precision, the specific

167 Article X of the Federal Rules of Evidence, particularly Rules 1001-1004. See: http://www.law.cornell.edu/rules/fre/rules.htm

mechanisms at issue should be examined, and an appropriate Δ identified to limit false positives. Thus, a POset is formed so that:

$$\forall t_1,t_2 , |t_1-t_2|<\Delta \rightarrow t_1 \approx t_2$$

Recent work in the analysis of overlay patterns of disk writes shows that ordering of file writes can be limited by examining existing patterns of file storage areas on disk.[168] More detailed analysis of time sequencing from traces to validate digital time-stamps has also been done.[169] The key is to gain adequate experimental evidence to bound the value of Δ.

Determining the Δ may be done experimentally, and documentation indicates that substantial Δ values may be found. The NTFS file system has a documented access time resolution of 1 hour and NT FAT has an access time resolution of 1 day.[170] A different sort of deviation is expected from the EXT4 file system in Unix, which has been designed to allow configurable delayed block allocation and thus writes for efficiency and storage optimization so that files tend to be contiguous on disk.[171] Another effect is that analysis of overlay patterns associated with disk writes[172] may not be valid as a basis for comparison to log data for time frames within the write time window for these file systems. Examination of Windows time stamps indicated that clock drift and anomalies produced time skews ranging from a few to 17,000 seconds, and sequence errors from these skews cause event sequence errors under current algorithms.[173] Δ is not apparently fixed for a particular system, rogue values with large deviations appear, timelines produce sequencing

168 Svein Yngvar Willassen, "Timestamp Evidence Correlation", Presentation at IFIP WG 11.9 International Conference on Digital Forensics, January, 2008.

169 Svein Yngvar Willassen, "Hypothesis-based investigation of digital timestamps", chapter in Advances in Digital Forensics IV, Ray and Shenoi ed., Springer, ISBN# 978-0-387-84926-3, 2008.

170 http://msdn.microsoft.com/en-us/library/ms724284.aspx “Not all file systems can record creation and last access time and not all file systems record them in the same manner. For example, on NT FAT, create time has a resolution of 10 milliseconds, write time has a resolution of 2 seconds, and access time has a resolution of 1 day (really, the access date). On NTFS, access time has a resolution of 1 hour.”

171 http://kernelnewbies.org/Ext4

172 Liu Zhi jun; Zhang Huan guo, “Time Bounding Event Reasoning in Computer Forensic”, 2007 International Conference on Computational Intelligence and Security Workshop.

errors, and Δ changes with time differently in different systems of the same type. In networks, different skews from different systems may be mixed in logging servers, producing still more complications. In recent observations of cellular phone times, devices in the same network reported skews on the order of 30 seconds from Naval Observatory time and from each other.

Time sequence analysis in unstructured content

For unstructured content, there is often content that is only available through non-digital examination of the DFE. For example, the placement of hands on clocks in pictures, shadow length and direction, sequences of changes in appearance or behavior, echos in rooms, and other similar indicators may reveal time-related data.

Sourcing and travel patterns

When content is transmitted through space, traces of its travel may result. For example, sets of message headers present may be used as indicators of sourcing or travel patterns and processing en route. Content from headers like "Received:" headers may be used to generate a tree indicating how many of a set of messages come through each of a set of paths.[174] This reveals information about the infrastructure in use that can be tested for consistency against other indicators, such as the "Message-ID:" fields corresponding to those messages or events. Comparison between different message headers, such as sourcing linked to identifiers, leads to internal consistency measures of the traces, and can be done by creating pairs associating characteristics of one header's content against another. As soon as a pair indicates a different outcome relative to the paired characteristics, an anomaly is detected, and internal inconsistency is demonstrated.

Many other content types hold sourcing and travel pattern traces. For example, Microsoft OLE files and many other document files often have substantial information related to sourcing and travel. The "last 10" area of OLE files historically contained information

173 Bradley Schatz, George Mohay, Andrew Clark, “A correlation method for establishing provenance of timestamps in digital evidence”, Digital Investigation 3S (2006) pp S98-S107.

174 F. Cohen, "Issues and a case study in bulk email forensics", Fifth annual IFIP WG 11.9 International Conference on Digital Forensics, 2009/01/27.

about the last 10 user identities to alter the files, but this has proven unreliable in legal matters. The date and time information contained within the file meta-data of the OLE files contained within these documents provides some sequencing information about how the document was assembled from parts, and the two file I/O times indicated, at one point in time, the date and time of the last output according to the system making the output and the time according to the file system that the output was written to. This indicates travel and sourcing when the local disk is not used to store the file, and can provide network file timing details that may be correlated to anchor timing information to individualize a file server used for storage. Other similar embedded content types may yield similar information about the origins and lineage of the overall document. For example, many such documents provide change records associated with multiple authors and writers and provide different writer traces and related date and time traces for changes made throughout the document's lifetime.

The generation of arrival trees is an $O(n^2/m)$ time and space process where n is the number of hops all messages combined took, and m is the number of messages. To show how this is done, consider creating a tree structure in which each entry for a particular field in a particular header (e.g., a "from" field within a "Received:" header) is read in order; with the "distance" in hops from the final destination and the ascension number in the message file used, respectively, to indicate the tree depth; and appended to a list of messages with the same tree up to that point. This entry is added to the existing tree branch, if one exists, or used to create a new branch if it does not previously exist. Each list can be implemented as a linked list with end pointers so as to make addition of entries constant time. Since each entry is added once per hop leading to it, the space and time consumed is $O(d \cdot n)$ where n is the number of header entries and d is the average distance in hops that a message travels in terms of entries created. For a collection of m messages, $d=n/m$, so the result is $O(n^2/m)$. While in theory, this may seem large, in practice messages travel an average of only a few hops before arrival at their final destination, so the actual time is reasonably approximated by $O(d \cdot n)$ where d is a constant. Which is to say, the time and space are linear in n, except for some pathological case where there are far more hops

per message than there are messages. That case alone would be an indicator of inconsistency with historical experience and the operation of messages in the Internet today. This is readily contrasted with a typical word processing document where a smaller number of files typically go through a larger number of modifications, often involving many authors.

Presentation of the resulting tree may be revealing in terms of associating actions to actors, and thus it can be used to find consistencies and inconsistencies with entries. Tree insertion can be made O(1) in time by the use of no more than m hash tables of size 2n (O(2·n·m) in space), however for large collections, a time space tradeoff might be advisable to reduce space at the cost of time. Such a tree may be built for any traces of fields within message headers or documents, as long as they have multiple indicators in sequential order.

In the case of emails and usenet news posts, this is typified by "Received:" headers, but a tree can also be made based on the entire sequence of available fields or any subset thereof. Each of these are similar complexity. An example of this is given for email messages,[175] but similar analysis for documents and other sorts of records has not been published as far as we are aware. Clearly, some such analysis may be possible if the raw data is available, but the forensic utility of this information for other sorts of content will likely be very different than for messaging traffic.

Sources, destinations, parsed subsets, or other traces may also be used in conjunction with timing information to identify such things as performance effects and other similar damage-related issues asserted to be self-identifying from traces. For example, looking at time differentials for hops based on "Received:" header date and time stamps leads to traces of times associated with hop sequences that can be measured against volumes to determine whether, or to what extent, delivery times are consistent with claims of interference.[176] In this case, the generation of the relevant data is easy, but finding a process that might determine the effect requires some sort of averaging or other consolidation of traces and analysis

175 F. Cohen, "Issues and a case study in bulk email forensics", Fifth annual IFIP WG 11.9 International Conference on Digital Forensics, 2009/01/27.
176 Ibid.

of a relationship between the events claimed and the traces. For non-automated movements leaving fewer traces, this becomes less usable because it is less predictable.

Correlation of delay time associated with message delivery and volume of messages might be a sound approach, but given only the set of messages from one party to another, there is insufficient data to demonstrate what, if any, other messages might have been affected or had effects on the delivery times of the messages. In short, these traces on their own are inadequate to make a substantive determination of causality unless other outside circumstances provide details that allow such determinations to be made more authoritatively. Providing some such other data might be problematic from a privacy standpoint. While audit trails might provide indicators, analysis of audit trails in the absence of other content is also unlikely to be definitive.

Consistency checks across related records and traces

Similarity of time and other characteristics and features across headers, bodies, content, files, images, and other sources, such as system, process, or software logs, may also be used for consistency analysis. An excellent example of this is comparison of date and time stamps from meta-data or audit records to time and date stamps.

As an example, the "mbox" format[177] includes a "From " separator between emails generated by mail transfer agents (MTAs) sequentially as emails arrive. The date and time stamps of these entries should be sequential through a mbox file as delivered. If this is not true, except for file locking delays and similar parallel processing possibilities described earlier, it presumably means that the mailbox file was not original writing by the MTA. Rather it must be the result of some other process. If the date and time stamps of the "From " separators are substantially different from the date and time stamps on the final "Received:" headers, it indicates that the separator was put in place by a different process than the MTA reception mechanism. If other dates and times within the message correspond to the separator, a mail user agent (MUA), such as a

177 The mbox format is specified at http://www.qmail.org/qmail-manual-html/man5/mbox.html

mail reader, may have sorted the collection of messages in one way or another and then produced the mbox file. Some MUAs substitute new "From " separators or alter time information for that which existed in the original writing, indicating a date and time received, read, viewed, or used.

Any of these traces can be compared to system or software records such as the logs produced by the MTA, and inconsistencies between the times or ordering of these mechanisms are again indicators of other processes. When combined with events and identifiable characteristics of the operating environment and software mechanisms in place, these sorts of inconsistencies may go to the confirmation or refutation of trace consistency as well as event consistency. When mixes of these sorts of conditions exist within a single message file, its construction becomes even more suspect and inconsistent with original writing of messages as received. Type D inconsistencies between events and traces may also be sought by detailed questioning and/or interrogatories.

Comparison of separators to last reception headers, is O(n) with the number of messages. The more general comparison of any one to all other date and time indicators within the message is linear in the number of indicators once they have been converted to a standard format (i.e., normalized). The comparison of log files to content is somewhat more problematic unless some linkage between the traces can be established. This problem was identified in[178] and it was suggested at that time that the records include annotations linking to the traces produced by the programs, (e.g., MTAs or database programs) that produce them. Process identifier annotations, message identifiers, internal identifiers, and other similar indicators commonly placed in log files sometimes provide such annotations. Assuming that theses linkages can be made reliably, the time to compare traces, once associated, is linear with the number of entries after sorting, or $O(x \cdot n \cdot \log(n))$ in total, where x is the time to associate records. When identical indicators are present, x is a fixed time, and the sort is the limiting factor.

The comparison of all indicators to all other indicators is potentially more problematic if the assumption is taken that minor time

178 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

deviations are normal (i.e., $\forall t_1,t_2$, $|t_1-t_2|<\Delta \rightarrow t_1 \approx t_2$). Sorting helps to perform analysis, but no algorithms have been shown for this correlation as far as we are aware. Algorithms exist for the general sorting problem at complexity $O(c \cdot n \cdot \log(n))$ where c is the complexity of the comparison function. "Sliding window" techniques have been used for similar problems, but this area is relatively unexplored in digital forensics today. Statistical studies have not been published on the normal consistency of ordering or deviations in timing associated with particular systems, and such studies would have to be done with similar systems to those in question for validity of results to be demonstrated. This analysis has been done by reconstruction in legal proceedings.

Audit trails may be correlated to each other and the interaction of programs with other programs correlated to the audit trails to determine if they are consistent. This approach was undertaken in the 1980s.[179] Results indicated that creating false but consistent audit trails from existing audit trails is quite difficult.

In simple cases, known format for fields and records are assumed identifiable, and this is exploited to allow the analysis to be done efficiently. But complexity issues start to get more interesting as the traces are less constrained. For example, suppose all sequential traces are to be formed into a unified POset. A major potential problem is that traces may be generated in different ways. In one case,[180] adding and removing audit records and inconsistency in audit records were identified, both with respect to unexpected present and missing records in archives. But this study ignored the potential for audit records, meta-data, and related records, to have different time bases and granularities. If one program gets time data as it starts, and another as it ends, even though they start and end together, they may produce substantially different records. Internal ordering properties must be taken into account in such analysis, but only limited studies of such consistencies have been undertaken in the published literature to date. The value for the Δ identified earlier is harder to determine if different mechanisms are involved.

179 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

180 Ibid.

Anchor events and external bounding

In general, an anchor event is some event that can be asserted by the examiner based on personal experience or other similar authority and that can be linked into the issues in the matter at hand. As a good example, a time stamp from an email that the examiner personally sent from a system they personally managed that was kept synchronized with UTC via network time protocol was used to demonstrate that the time stamps from a third party service provider were accurate. This was then used to tie a set of messages to local times on other computers relevant to the legal matter.[181] Traces may generally be used to determine times relative to third parties if anchor events may be identified in the traces. Determining actual times or relative times by anchor events or differentials is problematic unless the anchor events can be tied to known authoritative time bases. The complexity of identifying traces that may be relied upon for anchor event linkages is linear for obvious traces such as those containing time stamps or IP addresses. This allows time frames of remote systems to be bounded with some level of credibility. Other sorts of anchor events, if understood by the examiner, may also be easy to find.

Time consistency between records will then help to provide internal trace consistency required to allow other traces and their associated events to be anchored to known external events. As the length of the linkage grows, the potential for challenges also grows, while more linkages may increase the probative value. For this reason, processes to identify known sources of linkages may be valuable to put in place. Consistency between internal and external audit records, such as file transfer and system logs on different systems, may allow times and content across systems to be anchored and type C and D consistency to be further tested.

Anchors may also include things like geographic location or time zone. Geographical location tends to be important in many legal matters because of jurisdictional issues and because of claims made by parties. A classic example is the assertion that a person or

181 F. Cohen, "Issues and a case study in bulk email forensics", Fifth annual IFIP WG 11.9 International Conference on Digital Forensics, 2009/01/27, appearing in "Advances in Digital Forensics V" I. Ray and S. Shenoi, Ed., 2009.

system is in one place, such as where some act was or was not committed, when there is DFE indicating that they were elsewhere. While there are many complexities associated with such a claim, clearly the traces associated with the DFE are critical to detecting the difference between proof of innocence and deception.

One trace analysis example is the comparison of time and time zone patterns as opposed to locations of devices. In message traffic, the format of date and time stamps often includes time zones. The time zones are typically placed into the header fields by the receiving computers as they add headers. As a result, the time stamps and zones must be used together to determine accurate time in UTC and then recomputed into local time in the relevant location. These transformations are relatively simple, O(1) computations, but they are prone to errors because of the complexity of relating dates and times across time zones, taking into account calendar variations of various sorts, and compensating for errors in time settings of computers. Time zones, however, typically remain stable within a single computer and application. If traces of an application executing within an environment show time zone deviations other than those normally associated with system, and application, inconsistency is indicated. Many conditions may cause this, and it is most damaging when tied to events like assertions that records reflected in the traces are accurate and that the system was in a particular location over a particular period.

Identifying such anomalies is readily accomplished by taking the sequence of time stamps that should relate to each other and identifying changes in time zone as indicated by time zone offsets in the time zone field of date and time stamps. Normalization to a particular time zone is O(1) as it only involves a simple addition or subtraction. Sorting by time stamps after normalization to UTC or another desired time zone is $O(n \cdot \log(n))$ just as the sorts above. Detection of changes in time zone involves only the direct text comparison of a few characters between one record and the next in the sorted list, which is complexity O(n) for a trace with n records. Selection of records where the time zone is shown to have changed is readily done and an effective presentation consists of the sequence of from-to periods of each time zone and the time zone change times. This may reveal inconsistencies between claimed

events and traces. Missing time zone indicators are also possible in these sorts of traces, and if different from other related records, their presence or absence may indicate inconsistencies with events, such as the assertion that records were made by identical mechanisms.[182]

Care must be taken in the use of tools to interpret times and time zones. It is generally preferred to use tools that display traces and their representation in different formats. Tools that "interpret" time and date information often produce errors.

Time differentials and jitter

Time sequences and differentials are also useful and may be computed in a very similar fashion. Time stamps are sequenced by the placement within sequences or data structures and differential times are computed by date and time subtraction. Date and time arithmetic is often computed by converting all relevant dates and times to time distance from a common starting time, such as Jan 1, 1401, or whatever the time reference is for the date and time formats in use in the environment. Relevant calculations are done in this time base and results are converted to current time frames.

Time differentials and translations are problematic for many cases because, for example, a date difference between Feb 27 and Mar 1 is either 1 or 2 days depending on the specific year involved. Many have tried and failed to get date and time calculations right, and when time zones change along the way and differentials of seconds or minutes are at issue, special care should be taken in properly characterizing what took place. However, when UTC conversion is properly done, and dates and times are converted to time steps since a defined start date properly, time differentials are date independent, consisting only of a count in the time metric, typically microseconds, milliseconds, or seconds, depending on the accuracy of the records and the precision of conversion routines.

Conversion routines for date and time stamps can be implemented with complexity O(1) for each conversion done, and sequences are given by the original traces, so time for computation is linear in the number of time stamps being compared. The result is typically a

182 See "International Workshop on Learning Classifier Systems (IWLCS)", an annual conference on learning classifiers and related methods.

series of delays, most of them positive (time moves forward), and sometimes some of them negative (due to differentials in time settings or inconsistencies in time zone settings).

Taking this process a step further, when the same computer is involved in more than one sequence within a trace, the time differentials of that computer relative to other computers may be analyzed to detect consistency or inconsistency between time stamps and behaviors indicated by traces across computers. For example, if a message delivery process typically involves a particular sequence of computers and each places a "Received:" header in the messages as they are processed, the time differential between computer-to-computer times should and, based on limited samples, usually does indicate some level of consistency in time differentials.

Detection of substantial changes in differentials or differential time anomalies are inconsistencies with the notion that the same process was applied to each of the messages. Many possible causes exist for these sorts of differentials, and depending on the nature of the issues at hand in the case, these should be investigated to make more definitive determinations.

The complexity of doing these sorts of machine-to-machine differentials of records across machines is O(n) where n is the number of time indicators used, assuming that the time indicators are reconcilable to a common format in O(1) time. But the correlation of these traces may be far more complex, depending on the nature of the correlations being done. Typical approaches such as gathering statistics on means and deviations are O(n), but these are not particularly well suited to the types of errors that occur in digital systems, which generally fail in step functions rather than having deviations from a norm based on random stochastic processes.

In cases we have seen, there are many instances of messages delayed by days purported to be delivered by the same process that, during the same time frame, delivered seemingly equivalent messages through the same paths in a matter of seconds. In one example, a message was delivered a second time after 6 months of

delay from its original delivery.[183] This is an inconsistency with "normal" behavior, but did not, in this particular instance, indicate anything nefarious. It appeared to be the result of a restoration from an old backup where residual data from the MTA triggered a resend of an old, already sent message.

Even within a file, time differentials may be indicative of other related issues. One such example was a case in which time differentials between normally identical date and time stamp indicators within an object linking and embedding (OLE) formatted file were potentially probative with regard to the matter at hand. In the process of analysis, reconstruction was undertaken and it was determined that the time differentials were associated with differences in local and server time bases. These differentials were themselves then available to use as an anchor event for further analysis.

Issues of base rates and assumptions in analysis

In examining things like event sequences based on traces, the notion of consistency relative to causality are sometimes problematic because of added assumptions made in the analysis. For example, in analysis of illicit file access,[184] an assumption was made that access to files by a user should be between the time of their login and the time of their logoff. Thus, when a user is not logged into a system, they should not be accessing files. A detailed technical analysis showed that false positives were detected in experiments. After tracing this down, it was found that the sendmail daemon acted as the user, under their user identity, when working on the user's files as part of sending and receiving electronic mails. This meant that when the user was not present, the computer was acting on their behalf, even though they were not logged in, and of course this particular access was not illicit at all.

The problem comes in identifying all such situations and performing analysis to differentiate between normal and abnormal accesses of

183 F. Cohen, "Issues and a case study in bulk email forensics", Fifth annual IFIP WG 11.9 International Conference on Digital Forensics, 2009/01/27, appearing in "Advances in Digital Forensics V" I. Ray and S. Shenoi, Ed., 2009.

184 P. Gladyshev, "Formalising event reconstruction in digital investigations." PhD Dissertation; University College Dublin; 2004-08.

this sort. If we eliminate all sendmail activity from the analysis, we may miss many known sendmail problems that would be associated with illicit access, but determining all of the cases under which such access is or is not appropriate is problematic.

Generally, this sort of problem falls under the statistical notion of base rates. In essence, when looking at behaviors, we need to subtract out base rates if we are going to start to draw statistical conclusions. If the normal behavior of a system is that sendmail acts for a user when not logged in, and if this normally happens at a rate of files accessed for each email sent, normally varies by a particular amount, and if the behavior appears to be that of a normal distribution, then to detect a statistically valid anomaly, we need to determine that the rate of such changes exceeds the base rate by more than a defined number of standard deviations to make a prediction that it has a particular likelihood of being a true positive as a detection of illicit access.

Unfortunately, many such behaviors happen only once for a given illicit activity, and many such activities don't happen often enough to be statistically identified. Perhaps more importantly, from a forensics standpoint, a statistical likelihood that some such access was illicit is likely to be poor at best in associating a particular criminal or civil violation with the acts of an individual. Thus it all seems to come down to particular cases when seeking to meet a legal standard of proof.

Quick summary of characteristics and features

Characteristics and features are extracted from traces based on typing analysis and assumptions. Checks are made based on the characteristics and features to determine type C and D consistency and inconsistency, and results used to probe issues in the case.

Building sieves and counting things

Much of the work of forensic analysis consists of the examiner building sieves to extract specific derived traces from other traces, and counting things within the original or derived traces.

Extracting derived traces from other traces

The extraction of derived traces from larger traces is exemplified by the search processes described above. But in building sieves, the

examiner is creating new tools from existing tools for the specific purposes of the case at hand. While examining a minimal trace, such as a sentence or a few pages of text, may be within the realm of manual activities, traces in modern cases often involve millions or billions of bytes. Manually working with these large traces is impractical because of the time involved and unreliable because of the large number of things that may occur within them. Even a log file from a server such as a Web server may collect millions of entries per day, depending on the popularity of the Web site and the detail level of the logging mechanism.

The extraction processes used for many cases starts with standardized approaches, such as the extraction of emails from an "mbox" formatted mail collection, or the parsing of a Web server log to extract universal resource locators (URLs) from it. But at some point, the examiner has to identify specific information related to the case and search for traces in order to determine consistency. For example, if the case is about messages containing a particular URL within Web pages, the examiner typically selects out only the relevant parts of the traces that contain the URLs in question and performs further analysis on those extracts. In a Unix-like environment, that might be done by a process using the "grep" command with a regular expression specified, taking the larger trace as input, and producing a derived trace as output. The result may be stored in a file on the examiner's system or it may be directly sent to a further analysis step using a "pipe". The results of this process may then be reviewed by the examiner to determine whether it produced the right derived trace, and further sieved to get a further derivative. At some point, the analyst may determine that the result is usable for the purpose, and go from there.

This process is often repeated in multiple rounds working with different tools, specifications, and mechanisms, to produce different sieves. If the process is to be repeated many times it may be placed into a computer program, either as a "script" or in some other form, and named for later use. Over time, this sieve may be tested more thoroughly and validated, and become part of the examiner's toolkit for future uses. But it is likely that it will have to be modified with time as its uses change.

Building and using derived traces

In the process of examination, because of the way computer programs work, it is common practice for examiners to derive traces from other traces. This process is not always just the extraction of traces from other traces. It may involve creating sequences that never existed in the original trace, associating the new sequences with previous traces, and working with the derived traces to seek consistencies and inconsistencies.

The problem with derived traces is that analysis of derived traces directly provides information about the derived traces, but only indirectly provides information about the original traces. As a result there is at least one additional step that has to be done in providing the results. That step shows that the derived trace properly relates the meaningful information about the original trace through the derivation and analysis process, so that statements made based on the analysis of the derived trace apply to the original trace.

A typical example of a simply derived trace is:

- extraction of syntax elements from a trace, such as headers from message traffic,
- reformatting of the syntax elements to normalize, such as combining header continuation lines to leave one line per header.
- sorting, such as by extract number or field values,
- numbering, like formatting each line to include the sequence number, the line number within the extract, and separating results into files named as the extracts start.

This process is designed to allow the derived traces to be related to the original trace, but it also allows automated processes on the derived trace to be done more easily, such as:

- Sorting by entry number within the derived trace,
- Extracting particular numbered entries.
- Performing path analysis to determine the paths by which the extracts were sent in their journey from place to place.

- Calculating time delays and differentials associated with each of step in the journey

All of this is done while providing for association back to the original traces.

A more complex example of a derived trace is the creation of a parallel coordinates graph that depicts the multidimensional space of complex data sets, such as log files, in a 2-dimensional plane. This has been done for log files and other similar data sets[185] to provide accurate depictions of traffic flow patterns in a graphical form, but is limited in practical use to volumes in the thousands of relevant traces because visualization fails in larger volumes.

Derived traces are useful in performing analysis, but the resulting statements about the original trace must be done with proper caution, and the examiner is well served by finding an independent way to validate the results after they are known.

Counting things

Another very common process is the process of counting things. It should be no surprise that counting is something that computers are used for in DFE examination, but counting is not all that easy to get right, even when a computer is used for the purpose. While in general, computers are far better at counting large numbers of very similar things than humans are, computers are problematic, even at this seemingly trivial task.

Many analysis programs and utilities used for counting things, use the inherent representations of integers or other numerical storage methods of the underlying computer. These are predominantly fixed-length fields containing 16 or 32 bits, with some modern machines using 64 bits instead. A 16-bit integer in the one's complement representation that can only store positive integers, can count at most to 2^{16}-1, or from 0 to 65,535. A 32-bit value that is allowed to contain negative integer values will be limited to 2^{31}-1, or 2,147,483,647. Depending on what is being counted, even 2 billion is not necessarily large enough to hold the result. For

185 S. Tricaud and P. Saade, "Applied Parallel Coordinates for Logs and Network Traffic Attack Analysis", 18th EICAR Conference, May 9-12, 2009, Berlin, Germany.

example, a 1 Terabyte disk has about 10^{12} bytes, far more than the $2*10^{10}$ maximum count that is available from a 32-bit counter.

The result of a count that exceeds the maximum stored number value may be an error message, or it may simply be a wrong number with no other indication. The analyst must be knowledgeable of such limitations and careful to assure that they are properly handled in the use of tools.

Some tools, like the "Lisp" language and the Unix "bc" utility, inherently support big numbers, which are numbers with unlimited precision. If properly used, this provides the means to eliminate all of the obvious counting problems associated with maximum integer values. But this is only the beginning of the challenges faced by the examiner, even in trying to count things.

Combining mechanisms and dealing with resulting errors

In many instances, examiners must combine sieves, counts, and derived traces to get to an answer, and in some cases, this can include a substantial amount of custom coding. Examiners with adequate skills to do this level of digital forensic examination, typically write small programs in the course of an examination. They build up a library of such tools that they use over time and combine them with new or altered variations to produce ever increasing sorts of results with less and less effort per useful answer.

Notionally, analyses that are more complicated have more potential for errors. In practice, an error at any step of a complex process may result in process outputs that are incorrect or cause the process to fail. As sieves, counts, and derived traces are combined into more and more complicated instruments, these instruments tend to become increasingly fragile. For example, if the syntax of a regular expression works, but fails to cover all of the possibilities for realizing the identified thing being sought, or perhaps identifies things that are not being sought but that fit the defined pattern, the number of instances reported may be wrong. These are false negatives and positives, respectively. If results of a sieve produce false positives that are not detected during testing, and the result is sent to a counting mechanism, the resulting count may be higher than it should be. If there are false negatives, the count may be lower than it should be. If there are both false positives and

negatives, the result could even be the correct number, even though it was derived the wrong way, and the resulting pointers back to the original traces might be wrong, even though the count is right.

While some may wish to assert that the existence of false positives and negatives are the result of sloppy examination processes, the reality of examination processes as they exist today and are likely to continue to exist into the future, is that they are complex, error-prone, and very hard to get "right". It's often hard to even clearly define what "right" means in a way that is useful to the process.

In a DNA test, a known sample is compared to a new sample in a very well defined "syntax" of the appearance of spectral lines at particular places on a calibrated measurement output to find a match. Unlike a DNA test, searching DFE (1) often involves syntactic elements that are not well defined and may range over a far larger space, (2) often involves far greater volumes of data, (3) is sometimes designed to create problems for the examiner, and (4) is almost never designed to support the forensic examination process.

Those who assert that perfection is attainable have likely never had to define a parser that correctly parses all formats used by all existing and future processing mechanisms, and produces mappings between all formats used for all fields and all other formats used for similar fields. Such a program would have to take into account all of the programming errors made by the people who wrote all of these programs, all of their interpretations of specifications, every possible way in which a malicious or accidental actor might have produced output, and properly sort out all of the differences.

Perhaps the only reasonably workable solution to the increased fragility and error modes of more complex tools is the use of more redundancy to detect and compensate for such errors. This redundancy may come in many forms. While increased testing is also a useful approach, the schedule of most legal matters prevents extensive testing of examiner-generated mechanisms. Building each step in the process to self-validate is not feasible and would not prevent many of the sorts of errors that result from specification

limitations. Two common forms of redundancy that seem to work well are (1) using independent methods to generate results and comparing the results, and (2) checking intermediate results to verify that they are correct so that errors do not accumulate through the process.

Each of these has its limitations as well. In using independent methods to verify results, how independent must they be? Must they be in different operating environments? Must they use different command sets, programs, hardware, software, input and output devices, and examiners? In checking intermediate results, does each intermediate step have to be checked? What constitutes a check adequate to eliminate all false positives and negatives? How do we compensate for errors in the checking mechanisms?

Checking also consumes resources, and since resources are limited, the examiner must ultimately choose between additional verifications of their results and generating other results. Wrong results that are caught have high impact in that they destroy the credibility of the examiner and their process as well as the specific result that is wrong. While a single trivial error won't likely eliminate the probative nature of all of the work of an examiner, the examiner who makes many such errors may end up finding that they are unable to find further work in the field for 7 years, a typical period over which records of previous mistakes are available for court.

At some point, it is necessary to give up the checking process and decide that the results are "reliable enough". But with no way to measure this reliability, an uneasy feeling may remain. Perhaps the best solution is to allow challenges by the other side. And thus the notion of the oppositional legal framework of the US.

Finding things that are intentionally hidden

Analysis is commonly used to try to find traces reflective of content that has been intentionally hidden. Intentional hiding techniques generally include deletion of content, transformation of content into forms that make it harder to recognize and comprehend, and placement of content in locations where it is not normally found.

In general, the problem of detecting and finding intentionally hidden content is not solvable, in the same way as it is impossible to

definitively determine meaning. Meaning is a function of intent, and human intent is in the mind of the individual, not in the traces or the mechanisms of DFE.

In essence, finding "hidden" content is no different from finding any other content. It typically starts with type identification, which, if done without accounting for the potential for multiple simultaneous types, may seem consistent with one type while it is in fact also consistent with other types.

Deletion and placement in hard-to-find places

Deletion of content comes in two general varieties.

1. Most deletion involves a user command or button press that the computer interprets by making removing pointers to content previously available under a certain name or through a particular path. The bits corresponding to the pointers to the content from elsewhere may be removed, replaced, or redirected. The traces remain, but they are not as obvious because the pointers are altered. So-called undeletion may often be possible, depending on whether the system supports it and whether the traces have been overwritten by subsequent activities.
2. The less common deletion method involves overwriting the original traces and/or the entries that point to them. In this case, the traces will no longer be available in digital form, but there are common cases where partial traces are not completely overwritten or where previous versions leave traces because they were not deleted in the same way. This may include, without limit, residual traces from memory, temporary files used by programs that access or manipulate the content, traces left in search engines and databases, and portions of the content shared with others, attached to other things, residing in logs, or kept by other mechanisms.

If the traces do not remain in digital form, analysis will not find them. If traces do remain, they can be found, but they may not be recognized. When S and ℛ support it, examiners make searches of areas marked by files, file systems, systems, databases, and other mechanisms, as unused, deleted, or otherwise unavailable.

Content may also be hidden in places that are "hard to find", such as areas of files or file systems not normally used. Searches that fail to find such content typically do so because they (1) fail to analyze portions of available traces or (2) fail to properly or completely type traces. The solution is to spend more resources in analysis and either make fewer assumptions or examine them more closely. Assuming that the traces are available, regardless of where the content is within them, it can be found with a thorough enough search. But it may not be readily recognized, and we have already seen that complete thoroughness is often infeasible.

Steganographic content and other transformations

The transformation of content into a form that makes it more difficult to recognize or comprehend is a broad subject area. Methods typically include coding, cryptography, translation, syntax matching, labeling, steganography, and representational changes. They interact with each other, and multiple methods may be used together.

- **Coding** is the manner in which content is represented. For example, the most commonly used codings for strings today are American Standard Code for Information Interchange (ASCII) and Unicode, but there are many other codings in widespread use, and far more codings that are possible. These sorts of codings are used for different purposes, including without limit, efficiency, historical compatibility, and ease of use. For malicious actors, add "failure to use them in commercial forensic tools". Even such simple coding as ROT13[186] produces misidentification and mistyping.

  Searching in different codings involves the same search and match methods that are used for any other coding, except that the FSM that does the searches has to search using the different codings. While parallel codings can be implemented and some search methods currently use more than one coding method at a time or allow for selection of coding method among a small set if so requested, in general, the problem of trying all possible coding methods involves all

186 ROT13 translates each character by 13 characters so that abcdefghijklmnopqrstuvwxyz becomes nopqrstuvwxyzabcdefghijklm.

possible interpretations. This is infeasible, as discussed earlier.

- **Cryptographic transforms** are commonly used in hiding, even when other hiding techniques are used. This has the advantage of making the content appear to be random in nature and defeats most attempts at type differentiation and search. Cryptographic systems and implementations have different characteristics and these sometimes allow rapid typing based on traces. Such systems are designed and intended to drive up the complexity of interpreting the content, and thus the complexity of decryption is typically too high for practical purposes without the "key".

  The attacks on such systems are, most often, in the form of automated generation of keys and testing of those keys against the content to determine which keys produce useful results. Some such systems include indicators of successful decryption, and this helps in attacking them. Automatic key generation and testing using parallel processors, known plaintext methods, and hashing approaches, allow many such systems to be defeated by the examiner with adequate resources.[187] Cryptanalysis is a specialized field in and of itself.

- **Translation** into different languages or language structures understood by the participants, is often used, particularly by those who have a native tongue that is not the dominant tongue in the culture they live in. Automated translation into and out of different languages is also widely available, and is sometimes used for hiding content when the participants don't know a common language. Such translations have problems in that the translation from language A to B may not return the same text originally translated from language B to A, even if the meanings of the overall document may be similar.

187 M. Weir, S. Aggarwal, B. Medeiros, and B. Glodek, "Password Cracking Using Probabilistic Context-Free Grammars", 30th IEEE Symposium on Security and Privacy, May 2009.[This paper covers many available methods and recent research results in the field.]

Language detection is usually easily done. In most cases the examiner notices that the things they see are in a language they don't know. At that point they get a language expert to identify the language and translate. There are also automated translation systems, and they can be tried one after another to find a match that produces meaningful output. The Shannon information content density measure can also be used to identify and differentiate languages.[188]

- **Syntax matching** is a method whereby the content being hidden is encoded in such a way that it appears to be valid within a different syntax. For example, the word "help" might be matched with the hypertext transfer protocol (HTTP) "get" syntax used for Web page requests thus producing a Web request such as: "get /h.d/e.d/lp.jpg HTTP1.0", which is a valid request that embeds the hidden content as the characters between the "/" characters and the "." characters. Any desired content can be hidden in a series of Web requests of this sort. Other methods of encoding in matched syntax may also be used. This syntax matching can be combined with other methods, such as the mapping of characters into other characters, character sequences, and so forth.

  In general, detection of syntax matched content is as hard as detection of any other encoding, but there is often an advantage in that, as larger quantities of content are sent using such matching, the meaning of the content with respect to normal usage may seem deviant, and the rest of the exchange may seem a bit strange unless it is well thought out by the individual applying it. Syntax may be matched to any language in use, including programming languages, temporary file encodings used as intermediate files in automated processes, log file entries, or anything else that can be thought of.

188 Baden Hughes, Timothy Baldwin, Steven Bird, Jeremy Nicholson and Andrew MacKinlay, "Reconsidering Language Identification for Written Language Resources", 2006, European Language Resources Assoc. Available at http://www.cs.brandeis.edu/~marc/misc/proceedings/lrec-2006/pdf/459_pdf.pdf

- **Labeling** is sometimes used to misassociate content with type. For example, content may be placed in a directory normally used for other things, given a name typical of other content, and have internal markings similar to other content.

  For content that is known in advance, such as widely distributed software and related files, operating system components, known graphical images, and other similar content, there are substantial databases available for identification.[189] Between these databases and the JDLR-type mechanisms,[190] many types of mislabeling can be readily detected, but this only speaks to the consistency of labels with content to a limited extent. In general, there is no mandate as to how content must be labeled within systems, other than the requirements of the FSMs that input, process, and output that content.

- **Steganography** typically involves the embedding of content within other content. There are an unlimited number of different ways to do this sort of encoding, and there are scores of openly available products that implement steganography. It is easy to implement custom software for this purpose as well. The complexity of finding all possible instances of steganographic content is unlimited in the sense that any sequence of bits might mean or encode anything, depending on the mechanisms used to interpret it.

  Detection of steganographic content from traces typically involves searches for markers left by embedding programs, inconsistencies between content and typical content of similar type, and identification of other traces indicative of known steganographic software. Consistency with events, or the lack thereof, is also used to drive the examiner toward a search for steganographic content.

- **Representational changes** such as file system versions of everything, database content, database representations of

189 See: http://www.usdoj.gov/ndic/domex/hashkeeper.htm for detailed information on the hashkeeper database and analytical tools.

190 F. Cohen, "ForensiX", The ForensiX Just Doesn't Look Right (JDLR) mechanism is detailed in the source distribution available in http://all.net/ForensiX/Forensix.tar

file system content, and more generally, any other sort of representation of any other sort of thing, can lead to concealment of content based on the view taken of that content. For example, a document may be encoded as a set of files, with the file names representing each of the words within the document, and the ordering and appearance of those words indicated by integers placed within each file. The file system representation of this document would be hard to automatically associate with the content, but when visualized through an editor that stores the information in this form, the content becomes apparent.

Mechanisms that track search results, such as the Spotlight program that runs in the OS/X operating system or Google search engines used within some systems, may be viewed through an analytical process, to piece together the documents reflected in the search mechanisms, based on the pointers locating that content. Similar methods may be used for analysis of traces given that the representational scheme is understood by the examiner and automation is available to implement that scheme. But again, since there are an unlimited number of such schemes that may exist, and since available schedule limits the things that can be done, such efforts cannot be universally applied and may not be effective given the specifics of the representations used.

Recursive embedded languages

Because, and in the same way that, a universal Turing machine can implement any other Turing machine, including another universal one, each language can embed other languages within it. This can be done recursively, with the only limit being that the total available size of the trace combined with the efficiency of the embedding, limits the level of recursion and the quantity of content that can be embedded.

For example, within an ASCII document of 100 words, in an embedding that embeds numbers within the document by using the last letter of each word to indicate an octal (base 8) digit from 0 through 7, a maximum of 100 octal digits can be embedded. If that

embedding is, in turn, used to embed the EBCDIC[191] code for letters, the 300 bits of content within the 100 octal digits can contain only 37 EBCDIC characters. If these characters embed user identities and passwords, it is likely that only a few such user identities and passwords will fit in the 37 available characters. While this embedding is recursive, it is not recursive without end.

Commonly available methods of embedding allow any sort of content to embed any other sort of content with efficiency related to the specific coding method used. This specifically includes the use of the content hiding methodologies recursively, such as embedding encrypted content stenographically encoded within syntax matched to valid HTML within an object linked and embedded file contained within a document compressed in an archive and stored as an attachment to an email contained in an email folder within a file system that is embedded in an "mpeg" formatted movie file stored on a server available over the Internet via Web access. The most common methodology for storing large volumes of hidden content today is embedding of content within relatively larger unstructured content such as music, video, or graphical image files.

Indicators

The vast potential possibilities for hiding of information lead to potentially unlimited analysis time and effort. To reduce the time and effort, at the cost of missing many potential sources of such information, examiners typically look for traces or events that *indicate* the potential presence of hiding methods before searching for specific mechanisms.[192] Typical indicators include, without limit:

- **Searching for tools used for hiding content.** Software on the market indicates that it is capable of detecting more than 500 different programs that hide content, but this is only the beginning of software that can be applied to this end.

191 IBM's Extended Binary Coded Decimal Interchange Code (EBCDIC), is one of the widely used 8-bit industry encodings. Detailed information on EBCDIC can be found in the IBM publication IBM Character Data Representation Architecture, Reference and Registry, SC09-2190-00, December 1996.

192 P. Craiger, J. Swauger and C. Marberry, "Digital evidence obfuscation: recovery techniques." The Proceedings of the International Society for Optical Engineering, pp. 777-888, 2005. [This paper gives select examples of approaches and tools to detect limited steganographic presence.]

- **Searching for apparently similar things that are different at the level of traces.** For example, if hiding alters a graphical image and the same image appears elsewhere, any bit-level difference between seemingly identical images is consistent with information hiding.
- **Searching for inconsistency with continued normal use of the operating environment apparently present.** For example, if the normal operation of the system is periodically stopped and this is inconsistent with file dates and times, then this indicates that some other process was operating during the time the normal operation was not operating. This is consistent with an external mechanism for information hiding in the content altered during that time period.
- **Searching for traces produced by known software that uses information hiding.** This includes, without limit, traces of installation processes, traces in logs, traces in program registries, temporary files produced by the software, traces of external communications with registration servers, traces of downloads, and other similar things.
- **Detection of files with anomalous content or content indicative of steganographic hiding.** A typical graphical image has characteristics like monotonically increasing and decreasing color values across lines separating objects. Steganographic programs tend to use low order bits in locations like these to store content, and when this is done, it becomes recognizable in analysis. The information density of different sorts of files based on the Shannon content measure also tends to be higher when hidden information is present than when it is not. Steganographic mechanisms sometimes leave headers or other indicators of presence as well. All of these things can be used in analysis to indicate consistency or inconsistency with a hypothesis regarding the presence of hidden content.

Given that hidden content is consistent with a hypothesis or an event, the examiner may also try to decode the hidden content, depending on the nature of the hiding mechanism.

There are a wide range of commercial programs offered for finding hidden content in forensic analysis, but these tools are typically not fully vetted for forensic applications, and proper care must be used to understand their limitations for presentation in court.

Visualization and other cognitive methods in analysis

Visualization is commonly used in analysis to increase the rate with which the examiner can make decisions about and decide where to focus attention on similarities and differences. The use of the human visual cortex to differentiate things is far faster and more efficient than other methods, but it is also less precise than computerized methods. For example, in looking at a series of similar texts, flashing one after the other on the screen or putting them side by side, allows the visual systems of the brain to almost instantly match similarities and detect differences. The brain spots these things very quickly, whereas programming a computer to do the same thing at the same speed for a particular application takes a long time. For that reason, the tradeoff favors the human brain for applications that are not done repeatedly or are semi-custom.

Sonic information is also useful in human analysis of digital data, particularly when the data is a representation of sound. While matching a specific audio file to another may be simple if they are exactly the same, trying to use computers to listen to a lecture and determine what it is about is simply too hard for computers to do for most cases today. Again, the human cognitive system is very efficient for these tasks to the extent that they need to be done quickly and don't require extremes in precision.

The rapid application and reasonable precision of human cognitive analysis is greatly aided by the use of software to rapidly and repeatedly place the relevant content in front of the examiner's cognitive system. For example, a computer program can quickly find the obvious graphical images in a large storage device and display one after another for rapid detection of which of the images are relevant to the issues at hand in the case. So-called thumbnail depictions of images allows a constant size with a defined granularity to be presented for quick review. The examiner can then identify the images of interest for more in-depth study.

Another application of visualization for analysis is in the display of large data sets in a manner that allows particular characteristics to be examined quickly.[193] For example, a sector by sector view of the traces asserted to represent a disk drive will show used and unused areas, file types apparently in those areas, location of file allocation tables, and other similar data for a whole disk, on a single screen. A file can be shown in such a depiction, in terms of the locations in the trace where the file resides, and so forth. Depictions of file content with colors shows compressed or encrypted files looking largely like noise, while other file types have patterns associated with their characteristics, depending on the visualization mapping from content to display. Log files can be shown at very small font sizes with coloring to reveal different log sources and the pattern with which they lay down their logs over time.

Highly graphical visualization is most commonly used to "Explore" large volumes of data. For example, parallel coordinates[194] is rapidly gaining interest in the examination community because it provides a means to visualize high dimensional data. But at the same time, the ordering of the dimensions in this method produces dramatically different depictions. Depending on what the examiner is trying to learn about the traces, such visualization can be very helpful or a hindrance. The use of such tools requires that the examiner have an understanding of what they mean and how they are to be interpreted. Visualization can lead the examiner to explore different paths for analysis of the data by providing insight into what is located where and how much of what sorts of things are present. But this also means that the examiner may be misled by the visualization. There is a great temptation to spend time examining visual images and depictions, while writing a short program to sieve through a high volume of traces for some particular characteristics in some particular syntax is far less visually stimulating, and may be far less effective.

193 B. K. L. Fei, "Data Visualization in Digital Forensics", Masters Thesis, Computer Science, University of Pretoria, South Africa, 2007.

194 S. Tricaud and P. Saade, "Applied Parallel Coordinates for Logs and Network Traffic Attack Analysis", 18th EICAR Conference, May 9-12, 2009, Berlin, Germany.

Without visualization of some sort, it would be essentially infeasible to do anything in digital forensics. The character-based display of results of typing commands is a form of visualization too, and the output produced by analysis programs must ultimately be presented in some manner for the examiner to comprehend their meaning relative to the matter at hand. Selection of the outputs of the analysis process are as important as the process itself, in that, without the output, the process produces nothing.

Digital forensics tools today produce a fairly limited set of views, and are predominantly used to show structured views of content. Things like directory structures, structures within files of particular types, timelines, depictions of graphical images next to tree structured depictions of a directory structure, and other similar items are displayed in windows on the user's screen. Scrolling, drill-down, enlargement, or view selection is typically available to the examiner. Input, output, and analysis method options are also typically provided through the user interface.

These tools are summarized as providing imaging (collection), analysis, viewing, and reporting.[195] For analysis, they include file signature analysis, hash analysis, email analysis, registry analysis, filtering, and searching. For viewing, they typically include previewing and file viewing. Clearly, there is a long way to go in the visualization of analysis processes and outputs.

Because examination tools do not provide substantial support for applied visualization, examiners tend to identify and use a wide range of tools from the commercial and free software space. To view network traffic, a packet analyzer is commonly used; to view bit sequences, a tool like "hexdump" is used, to view text files, a program like "less" may be used, and so forth. Forensic fonts™ are now available for definitive visualization of byte sequences,[196] but this area has a long way to go.

To get a sense of the magnitude of the tools used for visualization of different sorts of content in different formats, the "White Glove"

195 B. K. L. Fei, "Data Visualization in Digital Forensics", Masters Thesis, Computer Science, University of Pretoria, South Africa, 2007.

196 F. Cohen, "Fonts For Forensics", IEEE SADFE (in conjunction with the IEEE Oakland Conference), 2010-05-19, Oakland, CA.

bootable CD-ROM used and sold for digital forensics, included about 500 different software programs just for seeing digital data in different ways. The standard Unix utility program suite contains hundreds of such programs, many of which are used daily by forensic examiners. And many of the program used for filtering and building sieves produce outputs that are viewed by examiners using many of these tools, depending on the specifics of the matter at hand.

Visualization is core to DFE analysis, and yet there is no strong scientific basis for determining which visualization method is most appropriate or cognitively most valid for any given situation.

Examples

Many authors have done work and demonstrated methods for digital forensic analysis. Some of them are discussed here in light of the general results of this chapter to bring context and clarity to these issues. Interested readers should review related articles.[197]

Farmer and Venema

Farmer and Venema[198] (F&V) give examples of search and related analysis methods. For example, they show methods to search disks for specific time and date regular expressions in Linux:

```
strings /dev/sda1 | egrep '^Jan  [1-7] [0-9][0-9]:[0-9][0-9]:[0-9]
[0-9]'| sort | less
```

```
ps axuw|grep syslog | while read a b c; do pcat $b | strings |
egrep '[0-9][0-9]:[0-9][0-9]:[0-9][0-9]' | egrep 'Jan|Feb|Mar|
Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec'
```

These search a file system, disk, or memory for particular formats of records. Each of these are linear time up until the invocation of the "sort" in the first example, which is $O(n \cdot \log(n))$ in the size of the search result.

197 See "International Workshop on Learning Classifier Systems (IWLCS)", an annual conference on learning classifiers and related methods.

198 D. Farmer and W. Venema "Forensic Discovery", Addison-Wesley, ISBN 104-5123010-9411940, online at: http://www.porcupine.org/forensics/forensic-discovery/

They also produce various other similar searches that look for other kinds of records, largely using the same methods, and usually executing in linear time with the size of the searched content up until the use of a sort which increases time to O(n·log(n)) in the size of the search result.

F&V do similar analysis of file system records. They typically search the tree structure of a file system, which is linear time in the number of files present, and typically follow this up with a sort, which is O(n·log(n)) in the number of files in the file system. This is then used to provide lists of file-system-related traces in time order of occurrence. Their "Coroner's Toolkit" is freely available software that includes many of these routines, including the "MAC time report" which lists sorted modification, access, and creation times of files according to the file system. As an example of the typical way that such things might be done, here is a Unix command that, in normal operation, searches the file structure from the root (/) and produces a listing of the creation, modification, and access times for all files:

```
find / | while read a; do stat -t "%Y/%m/%d-%T" -f "%Sc
%Sm %Sa %N" "$a"; done
```

Next is an example of a program that, in normal operation, compares the MAC times to detect out of order creation, modification, and access, under the most common assumptions about their ordering. Note that this example, like the previous one, produces output that has time granularity of 1 second. While it can be interleaved with other sorted output in a similar format, the difference in granularity makes sorting inexact, and orders may be inverted between causes and effects if they produce time stamps that occur within the same second.

```
find / | while read n; do
  stat -t "%s" -f "%Sc %Sm %Sa" | (read c m a;
        if test $a -lt $c; then
                echo "Access before Create $a $c $n"; fi
        if test $m -lt $c; then
                echo "Modify before Create $m $c $n"; fi
        if test $a -lt $m; then
                echo "Access before Modify $a $m $n"; fi
```

```
)
done
```

This program also uses time stamps that represent seconds since "Epoch" (the "zero" time for the file system seconds clock) while the previous one produces a YYYY/MM/DD-HH:MM:SS format. The former format is useful in sorting along with times from other sources, although it does not provide time zone information, which is also problematic in finding absolute rather than relative times.

F&V use tools like the Unix "grep" command for linear time searching, "strings" for linear time extraction of strings from files, and other similar tools for similar purposes. They apply various sorts of these results as well, which are also $O(n \cdot \log(n))$ time in the size of the list being sorted, link deleted files and information notes, and do similar sorts of actions, again linear time in the size of the file system or content examined. They examine the search of information nodes, process identifiers, and other operating system constructs in much the same way and with similar results, and do similar things with system logs, system call monitors, and library calls.

F&V also introduce the notion of "order of volatility" to guide evidence collection. Order of volatility is related to identification, collection, and preservation. In essence, it asserts that the most volatile traces should be collected first to avoid spoliation.

F&V also discuss file reconstruction with "lazarus", a program that extracts disk blocks, types them by content, and allows the user (and automation) to view the disk and assemble blocks based on various approaches.

Willassen

In "Methods for Enhancement of Timestamp Evidence in Digital Investigations",[199] a hypothesis-based approach is taken to finding causal relationships between sequences of activities (they call these events), based on testing consistency with traces by Cohen[200]

199 S. Willassen, "Methods for Enhancement of Timestamp Evidence in Digital Investigations", Doctoral thesis for the degree philosophiae doctor, Trondheim, January 2008, Norwegian University of Science and Technology.

200 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

and subsequently by Stallard and Levitt. [201] This is done by using a logic programming variant on predicate calculus. Cohen used trace redundancy to detect inconsistencies, while Stallard and Levitt used “predetermined invariant relationships” to “detect semantic incongruities”. Willassen uses invariants and causality relations to check a clock hypothesis for consistency with time-stamp traces. This approach replaces the "anchor event" approach (type D consistency) with an internal (type C consistency) approach or a consistency approach based on events hypothesized by the investigator (type D consistency with hypotheses). Of course these can all can be used in conjunction with each other.

Unlike Carrier's approach,[202] Willassen uses the notion of causality ("Time is a partial ordering" and "Discontinuous time" identified in Chapter 3) to assert that; if trace A corresponds to a trace with time stamp i, and trace B corresponds to a trace with time stamp j; then there is an invariant of the form "if i<j then A<B". If the traces indicate that B<A when the time stamps indicate that i<j, there is an inconstancy between the time stamps and the traces that violates the invariant. This is then leveraged as a fundamental principal to do analysis, for example, of the order in which data is laid down on disks, the order in which audit records appear, and so forth.

A hypothesis is formulated by the investigator, and the hypothesis is tested for consistency with the available traces, but these traces are closer to those of Carrier in that they are not typically analyzed at the level of bits, but rather at levels associated with the methods by which the hypothesis asserts that they are generated. For example, a disk write is typically at the granularity of a block rather than a bit. Causality is used to assert that a disk block that was part of file A and that was overwritten by a disk block from file B implies that A was written before B. The time stamps from the files gathered from file system metadata can then be compared to the overwrite patterns to determine consistency of time stamps with overwrite

201 T. Stallard and K. Levitt, "Automated Analysis for Digital Forensic Science: Semantic Integrity Checking", Computer Security Applications Conference, 2003. Proceedings. 19th Annual.

202 B. Carrier, "A Hypothesis Based Approach to Digital Forensic Investigation." PhD Dissertation; Purdue University; May, 2006.

patterns. Software implementing this time-stamp logic is available[203] and runs against NTFS file system formatted images.

The "happen-before" relation (written →) is transitive in that if A→B and B→C, then A→C, and asymmetric in that if A→B then ~(B→A). The result is a directed acyclic graph, also known as a partially ordered set. In essence, the approach is to examine all pairs of causal relations that can be identified and that have identifiable time stamps; and compare the time stamps to the causal relations identifying an inconsistency if a cause has a time stamp later than its effect. If A→B, then $t_A<t_B$. If there are time stamps for A and B and the A→B relation is known, this can be tested.

The example above of finding inconsistencies in MAC times is one of the things that this analysis is intended to show. A more complex example would be the mixing of time stamps from system logs with file times, and analysis wherein causal relationships between programs and files is undertaken. A still more complex example would piece together times from message headers with system logs reflective of the arrival times of each of the messages processed by the MTA, and perhaps a mix of these results with the user's saved files containing the messages as the user received them in their interface. Presumably, the process start time would be before the "Received:" header time, which would be prior to the end of the process, and prior to the time that the user program stored the resulting message file.

Unfortunately, the analysis of time stamps for consistency for such things as file systems is O(n!) complexity[204] where n is the number of time stamps associated with an item of interest. While files in many file systems only have a few time stamps in the meta data from the file system, most system logs have many thousands of time stamps, as do files containing messages, server logs, object linking and embedding files, and other sources of content. To do a thorough job of consistency checking of time stamps across all of the different sorts of time stamps is clearly infeasible for the foreseeable future, at least using the algorithms known at this time.

203 Code is available at: http://www.willassen.no/phd_thesis/implementation/

204 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

A less precise, but less time-consuming, algorithm was also developed, and is usable.[205]

Also unfortunately, Willassen's experiments with time stamps show that systems created with no attempts to subvert time-related traces, produced substantial numbers of false positives, while experiments with intentionally falsely dated files produced false negatives. This does not bode well for this approach overall, but it does not mean that it is universally problematic. For example, analysis of other file systems or certain parts of file systems may be very consistent, but this needs to be determined experimentally for specific cases. Thus the use of reconstruction, as discussed later, may be required to assure more accurate results.

Other comments on the use of time for trace consistency

An improvement can be made, in that causes take time to produce effects. To the extent that these times are identifiable, the time stamps should differ by at least that differential. This then goes to the complexity analysis discussed earlier. To the extent that known run times or complexity analysis may be applied, they can be used to determine how much time it would take to do a particular operation under particular circumstances, and this can be used in analysis.

This general approach can be taken further, and has been in particular matters. For example, and without limit, the (1) time to make changes to a disk compared to the time available to make those changes, (2) time required to make a set of consistent changes compared to the available time and computing power available, and (3) expected delays and jitter with network traffic going over long distances, have all been used in analysis to show inconsistencies with hypotheses or internal inconsistencies in traces.

These are all examples of the same basic logic, but with added values for the time taken to do things. If $A \rightarrow B$, and $\rightarrow$ takes at least time Δ, then $t_A+\Delta<t_B$. If there are time stamps for A and B and the $A \rightarrow B$ relation is known, this can be tested. The challenge is

205 S. Willassen, "Methods for Enhancement of Timestamp Evidence in Digital Investigations", Doctoral thesis for the degree philosophiae doctor, Trondheim, January 2008, Norwegian University of Science and Technology.

defining the $\rightarrow$ relation for all pairs of time stamps or traces and identifying the constraints on Δ in a particular situation.

This drives up the computational complexity considerably for a thorough examination of time stamps in cases where there are lot of time stamps near each other, and changes the strict ordering of time into a POset. This was not considered in Willassen's analysis, and may be the reason for inconsistencies in MAC times. This may also be used to compensate for time granularity differences, effects of file locks, and similar timing error sources. Loosening constraints on ordering may also reduce analysis complexity in cases where many records are in equivalence classes based on the defined Δ.

The false positives and false negatives in experiments demonstrate that these techniques are weak, because the systems they were used on are not effective at protecting the limited time stamp information investigated. It appears that those systems contain a variety of falsely dated files or files whose "creation" dates do not accurately reflect the time and dates at which they were first created within the computers they were installed in. This means that some other method, such as an anchor event, must be used to try to bound date and time information, or some other method of doing more definitive date and time stamp analysis will be required before such a technique will produce forensically useful results and demonstrate reliability as an analysis tool.

Another issue in the analysis of time stamps stems from the different ways and places they can appear. For example, one classification of traces might include, without limit: file system structures, file content structures, metadata from file systems, time indicators from logs, time indicators in embedded file structures, output from listings that include time information, application storage areas that produce results dependent on time, transfer information associated with other systems, indicators of mounting and removing file systems, date and time information in archives, and the settings of system clocks. Finding and analyzing all of these different sources of time information for inconsistencies is problematic, at least because it requires that we type the sources of the traces and do the syntactic analysis required to generate the list of different indicators, create causal relationship maps between all

of the different types of relevant content, and find a way to perform the analysis on more complex sets of traces.

ForensiX

ForensiX is a graphical interface combined with a set of software designed to perform select analytical functions under a version of Linux. It was released in the late 1990s and has been used in various legal cases. As an analysis tool, it provided two things that other similar tools did not provide at that time; (1) a detailed record of every action taken by the analyst, and (2) direct details of what is being done for analysis in every step of its processing. These two things are particularly useful in analysis because they allow repetition of all analytical steps and allow the analyst to alter the way the tool operates so that it works more like a programming language than like a limited function tool. In essence, the graphical interface takes user input and forms Unix command lines using scripts that come with the program. The user can then alter the commands and introduce new scripts as desired to extend functionality and customize actions, while keeping a detailed record of all actions actually taken. The Computer Online Forensic Evidence Extractor (COFEE) uses the same approach.

The analytical techniques of ForensiX are not exceptional. While it had the JDLR function that searches for mismatches between file content and name, this is increasingly common in today's tools. It also had a variety of functions like imaging of disks and network traffic, which other tools also have, but these are beyond the scope of this book.

ForensiX does common things; like searches, sorting, and selection; and these form the basis of the analysis activities that it performs. The place that ForensiX stood out, in its day, was in merging the command-line style of operation that many examiners use to program their analysis functions, with the convenience and ease of use of graphical interfaces. The knowledgeable analyst could carry out novel analysis while tracking the actions taken and results given. For retrieval and repetition. ForensiX was also provided in source form and applied widely distributed and used software included in Unix distributions for tens of years. This allows

the examiner to drill down into the specific functions provided and testify about precisely what tools were used and how they operate.

The Coroner's ToolKit

This tool collection is an open source set of tools that is particularly useful in "file carving" and similar operations. It provides, among other things, a mechanism for going through an image of media on a block-by-block basis and characterizing each block by type. The blocks are then stored and organized in different ways to allow each to be examined. For collections of blocks that look like parts of a file to be examined together, it allows for their assembly. It also provides a Web-based interface to view the results.

An analysis of a typical disk may take days for TCT to complete, and it reveals a very large volume of information, which includes all of the content in the trace. But because the analytical results are provided as files accessible from a Web browser, results can be examined as they are developed, through the Web interface that depicts different sorts on the content. This allows examination to start almost immediately, while results are developed over time, and results get augmented as the analysis is undertaken.

To quote from the TCT Web site:[206]

> "Notable TCT components are the "grave-robber" tool that captures information, the "ils" and "mactime" tools that display access patterns of files dead or alive, the "unrm" and lazarus tools that recover deleted files, and the "findkey" tool that recovers cryptographic keys from a running process or from files."

The NIJ view of analysis

The National Institute of Justice has expressed a view on the analysis of digital forensic evidence.[207] The key parts related to analysis from the perspective of this book are included here. According to NIJ:

206 http://www.porcupine.org/forensics/tct.html

207 Forensic examination of Digital Forensic Evidence: A guide for law enforcement", http://www.ncjrs.gov/pdffiles1/nij/199408.pdf the National Institute of Justice, Special report, April, 2004.

> "Analysis is the process of interpreting the extracted data to determine their significance to the case. Some examples of analysis that may be performed include timeframe, data hiding, application and file, and ownership and possession. Analysis may require a review of the request for service, legal authority for the search of the digital evidence, investigative leads, and/or analytical leads."

Clearly the NIJ view differs from the narrower use of the term in this book, but their "analysis" includes:

- Timeframe analysis to determine when events occurred on a computer system and to associate use of individuals. This is typified by "Reviewing the time and date stamps contained in the file system... to link files of interest to the time-frames relevant to the investigation." and "Reviewing system and application logs...".
- Data hiding analysis to try to detect and recover such data and indicate knowledge, ownership, or intent. This is exemplified by "Correlating the file headers to the corresponding file extensions to identify any mismatches...", "Gaining access to all password-protected, encrypted, and compressed files...", and "Gaining access to a host-protected area..."
- Application and file analysis which includes; identifying files and programs that may be relevant to the investigation and provide knowledge of the system or users, "Reviewing file names for relevance and patterns... Examining file content... Identifying the number and type of operating system(s)... Correlating the files to the installed applications... Considering relationships between files [e.g., correlating history to cache files, etc]... Identifying unknown file types... Examining... default storage location(s)... to determine if files have been stored in their default or an alternate location(s)... Examining user-configuration settings... [and] Analyzing [metadata and content]."
- Ownership and possession includes; identifying the individual(s) who own, create, modify, or access content; knowledge of possession; presence at a computer at a date

and time; names used for filing and storage; and other related ownership or possession indicators.

Summary

Clearly the study of the use of redundant traces for consistency is only in its infancy, and the available methods for analysis and correlation of these redundant traces are already substantial. At a fundamental level, it seems clear that redundancy is key to consistency analysis, and that systematic identification and analysis of redundancy may lead to a more complete theory and practice of DFE analysis.

This chapter only covers the rudimentary forms of analysis in widespread use today. Further work is needed to characterize other classes of consistency checking in analysis, including, without limit; (1) Analysis of effects of parallelism, (2) Detection of similarity rather than more precise matches, (3) Addressing issues of mixed symbol sets and other similar environmental factors, (4) Analysis of possible consistencies and inconsistencies of missing traces, (5) Use of results to guide future events, and (6) Validation requirements for the methods used.

Questions

1. Given the legal requirements identified in this chapter, is it realistic to start with a bag-of-bits assumption? If we do start there, how far can we ever really go? If we don't start there, what will prevent someone from claiming that we erred in our original assumptions, so that our conclusions, no matter what they say, are not relevant?
2. Explain the sensibility of moving from a bag-of-bits to higher level models and how the assumptions associated with these models can be justified.
3. Feature and characteristic detection that moves from the bag-of-bits to the context of a computing environment seems to depend heavily on the knowledge, skills, experience, training, and education of the examiner. What of these do you bring to the table that could help you to identify features and characteristics that are not listed herein? Start listing the specific features and characteristics that you would look for in an examination to identify what the bag-of-bits is, and devise tests for them. How can you characterize the reliability of these tests, and how would you test the tests?
4. All of this computational complexity stuff seems like it is highly theoretical in nature. What is the value of understanding the complexity of these methods? How can this be used by the examiner in the real world?
5. How many symbol sets are realistically used today in computer systems? Given that most tools don't use the whole range of symbol sets, but rather are designed to be applied only to specific symbol sets and representations, how would an examiner handle traces using a different symbol set? How hard would it be for an opponent to simply use a different symbol set? How bad would this be for the examiner, and how would the examiner get around it?
6. Suppose the type of a trace is not identifiable, but there is no basis to believe that encryption or any other similar sort of method was used. What sorts of conclusions should the examiner come to? What is this consistent with and

inconsistent with? How could the examination proceed at that point?

7. Is it realistic to expect examiners to start writing parsers to do examinations, or if not, what tools are likely to have to be developed to avoid this sort of process?
8. What sort of normalization methods are you using today, and how do these methods point back to the traces that produced them? What sort of tool do you think you would need to make this really easy and effective? In doing the comparisons and searches with this tool, how would the tool be able to turn the examiner's notion of what is desired into a specification the computer could run?
9. How are imperfect orderings handled, and how do the information physics notions of finite granularity in time and space interact with ordering issues? How much leeway in ordering is permitted, and how much complexity in turning linear sequences into partially ordered sets will this produce under what circumstances?
10. To the extent that sourcing and travel patterns can be established for the movement of information as indicated by traces, how can this sort of information be used to establish causality? Or can it?
11. In doing consistency checks across related records, what sort of specialized assumptions must be made before the records can be properly related? How can the examiner be certain that these assumptions are in fact correct?
12. Anchor events require some reliable source of information, but how is the examiner to find such a source? And if such a source is found, how can the examiner be certain that the anchor source is accurate or how accurate it is?
13. For standard sorts of sieves, there might be software that implements the sieve with relatively little customization, but for more complex sieves, the examiner will have to program them. What sort of programming experience and expertise is required in order to program a sieve, and how can the

examiner verify that the programmed sieve does the right thing?

14. In building derived traces and doing analysis on them, what are the risks of getting wrong answers, and how can they be mitigated?
15. How can the complex interactions of different programs be accounted for in analysis?
16. For non-cryptographic hiding, where can traces conceal information that was intentionally hidden? How can these places be searched?
17. In dealing with recursive embedded languages, how deeply should the examiner penetrate the recursion, and is there a way to tell when there is no further to go?
18. How can visualization errors be avoided by the examiner?
19. The introduction of a Δ into time analysis turns a simple ordering relation into a far more complex structure, the partially ordered set. Review all of the mathematical analysis within this chapter and in the cited references, and identify the extent to which the introduction of this Δ would change the analysis of DFE. Does such a change alter the complexity results? Would it have a substantial effect on any of the cases you are aware of? How does it change the sorts of conclusions typically made about consistency and inconsistency of traces? Would it make analysis harder?
20. Download and try each of the freely available tools identified in this chapter on a disk drive such as the one you use every day for your work. Using those tools, see if you can identify DFE from your system that might be consistent or inconsistent with the work that you do every day. Assume you are tasked with showing that your system was not used for work purposes a significant portion of the work day. What evidence can you find that is consistent and inconsistent with that claim? Do these results produce false positives and negatives? How does this color your view of other systems?

6 Interpretation

In essentially every legal matter, there is a context for all traces. The context is the story behind how they came to be. Without such a story, they cannot normally be admitted as evidence, because there is no basis for establishing anything about their reliability or relevance. This story forms the overarching context for working on issues in the case, and it colors the thought and examination processes carried out in the case. The DFE examiner may not be given all of the details of the case, may even be intentionally shielded from some of the facts or contexts, and the opposing side will likely withhold some level of information from the examiner. As a result, the examiner must always assume that they do not have the "whole truth" or the entire context of the matter.

Occam's Razor is a philosophy of science approach that asserts, in essence, that the simplest explanation that fits all of the facts is to be preferred over other more complicated explanations, even if the more complicated explanations also fit the facts. While this principal is widely used in science as a whole, it it problematic as an approach for the DFE examiner because it is, or should be, assumed that "all of the facts" are not available. A trace or event may appear later that refutes the Occam's Razor explanation of yesterday, leaving only alternatives available as the truth of the matter. And not all traces and events are necessarily reliable, so assuming that they are reliable is also a mistake. The use of Occam's Razor as a principal is an example of an interpretation approach that is flawed in the DFE examination arena but that is commonly assumed and used in the greater scientific community. This is not to say that the DFE examiner should never use this principal, but rather that its use should be clearly identified as an interpretation and not treated as anything else.

These and other similar principles and issues arise throughout the work of the DFE examiner and are fundamental to the activities of the examiner. As such, interpretations constitute a process element in DFE examination that must be studied in order to be properly managed and undertaken.

Interpretation of traces and analysis results

Traces are analyzed using analytical methods, but they are also interpreted, in that the examiner chooses what methods to apply to which traces based on decision-making processes. These decisions about what to do and what not to do are a matter of interpretation. The analysis process that starts with a syntactic analysis to determine consistency with different assumed formats in the context of the case, such as the file system type, file type, parsing of a file or database, and so forth, uses interpretations by the examiner to identify what consistency checks to make next and which checks to ignore. The analysis of time stamps is based on the decision by the examiner to interpret the sequences that "look like" time stamps as time stamps. The analysis of delivery routes, time sequences, and all of the other analytical processes identified are all based on interpretations by the examiner. While these interpretations are usually not all that controversial, they may be wrong, and they may be right. We must recognize them as interpretations and not as analytical results. They are based on human or automated decisions and those decisions are based on interpretation and not pure analysis.

Keeping alternative explanations in mind

When considering issues related to analysis of traces, the examiner should always try to keep alternatives in mind. A common and appropriate practice for any attorney listening to or looking at the results of an examiner's analysis, is and should be to ask what other interpretations might be given to these traces, and what else could have caused these results. The examiner should always include caveats with regard to other possibilities, and those should generally include, without limit, relevant information physics items identified in Chapter 4 that might impact the results, explanations outside of the realm of the digital world, and the potential that deception or spoliation is involved.

Whenever internal trace inconsistencies (type C) are detected, an interpretation is called for. While it is certainly reasonable and appropriate to state something to the effect of "The analysis of [the identified] traces under the assumptions that [whatever the assumptions are] indicates that the traces are inconsistent in that

[explain the details of the inconsistency]", it is likely that such a statement will be met with further questions. For example, and without limit:

- Why is this an inconsistency? [the answer will likely ultimately be something related to information physics]
- Is the inconsistency because of one or another of your assumptions, and if so, which one(s)? [The answer is often something to the effect of a set of combinations of different assumptions that individually or in concert could be wrong, or deception, or spoliation. This is sometimes further resolved by a reconstruction or further analysis.]
- Can you explain how this inconsistency could have come to be present? [The answer typically comes by ruling out various explanations based on analysis or assumptions, and responding with whatever is not ruled out.]

All of these answers and the methods for producing them through the use of logic or other reasoning, are interpretations of the traces, and should be treated as such and stated as such.

Examples of trace interpretation

As an example of an interpretation, assume a case in which party A asserts that some set of "Received:" headers from a collection of message traces are authoritative and accurate as to the events that took place, and indicates what reception software was in use in their reception over the relevant period. Upon examination of these traces, it is found that, out of 100,000 such messages, 90,000 of them contained sequences of "Received:" headers in which the next to the last recipient was from a computer owned by party B and the last recipient was from a computer owned by party A. A reasonable interpretation of this situation might be stated as

> "The analysis of the identified traces, under the assumption identified by party A that all of these traces are accurate as to "Received:" headers, indicates that Party B sent 90,000 of the 100,000 messages to Party A".

This is clearly an interpretation in that the use of the term "sent" is not a formal term in the language of analysis and in that it draws the conclusion that the meaning of the "Received:" headers in the

traces is indicative of a message being "sent" from one party to another. It is also only partial in terms of indicating the totality of information about the history of the messages in terms of their paths from origination to destination, in that is does not state that party B originated the messages in question or anything else about their paths; only that Party B sent them to Party A. It also does not draw conclusions about the validity of the assumption about the accuracy of the traces, and indeed, "Received:" headers are easily forged or altered. It is thus interpretation and not analysis.

Interpretation and the presentation of statistics

Another example of interpretation, also taken from message traffic, is a case in which party A has provided 100,000 messages containing a variety of "Subject:" lines in their headers, and party A has also provided a list of claims relating to the meaning of each "Subject:" line for each message, claiming that some portion of the messages with "Subject:" lines are "mean". Without opining on the meaning of the word "mean" or the meanings of the words in the "Subject:" lines, the examiner may do an analysis of the claims against the "Subject:" lines. If they find that different messages with identical "Subject:" lines are claimed as "mean" or not claimed as "mean" with regard to "Subject:" lines, without opining on the meaning of anything, the examiner can still state something like:

> Out of 40,000 messages asserted to have "mean" "Subject:" lines, 20,000 of them have identical "Subject:" lines to other messages that are not claimed to have "mean" "Subject:" lines. Out of 60,000 messages indicating "not mean" "Subject:" lines, 12,000 of them have identical "Subject:" lines to other messages that are claimed to have "mean" "Subject:" lines. A's claims about which "Subject:" lines are "mean" are inconsistent.

This is another example of interpretation, at least in that (1) the indication of "mean" is based on the interpretation of A's claims as provided, (2) the interpretation of "identical" may be questioned or questionable, (3) there is interpretation of what constitutes a "Subject:" line, and (4) the assertion that these numbers indicate "inconsistency" is an interpretation of the analysis results.

An interesting question that is likely to come up with regard to such a statement is: "What percentage of claims are inconsistent?" The accurate examiner in this case must almost certainly answer something to the effect of "I don't know". When asked "Isn't it 40% inconsistent?", the answer has to be something like: "I can't conclude that from the available information."

These sorts of conclusions are illegitimate interpretations of these results because, at a minimum, they assume that all of the messages are present in the collection, an assumption that is not warranted. In fact, the messages are provided by Party A who is making claims about them. To the extent that Party A only provides messages that they believe are relevant to the matter at hand, other messages, including messages that may have identical "Subject:" lines to provided messages and that were not claimed under this or other categories of claims, might not be present. The actual numbers might be a million messages of which 900,000 have identical "Subject:" lines to messages in the 40,000 messages asserted to have "mean" "Subject:" lines.

But even if this were not the case, and all of the messages ever received were in the collection provided, the percentage of "inconsistency" is not a meaningfully defined statistical concept. For that reason, interpreting the results of analysis in a statistical framework based on this question is meaningless. In this example, there are false positives, false negatives, true positives, and true negatives, and all of these depend on the meaning of words about which the DFE examiner, who is not also a linguistics expert, is not qualified to opine. If you think this is being unnecessarily picky about semantics, the field of DFE examination may not be for you.

Unstructured trace interpretation

In unstructured data, interpretations get even more interesting. For example, the results described earlier indicate that repeatedly scanning the identical picture with the identical scanner without any changes to the configuration and in rapid succession yields substantially different traces. At this point, the DFE examiner who is going to opine on graphical images that seem very similar but that are not in fact identical at the level of traces, will be in the realm of interpretation, unless they state only that the images are not

identical. In similar fashion, images that contain indicators of time stamps, geographic locations, or other similar indicators, but for which there is no anchor event or other similar method for authentication, require interpretation in order to make statements that are likely to be probative. Since these elements are subject to alteration, this interpretation will need to be put in the context of the overall situation to assert or refute consistency.

Increasingly, DFE examiners are interpreting traces of graphical images in terms of angles of shadows, use of tools to modify them, whether they are originals or not, whether they are artist renditions or alterations of real photographs, and so forth. These are all interpretations and they are all problematic in that they ultimately depend on opinions and things like statistical studies that may or may not apply to the specific traces at hand.

The statistical analysis undertaken by Farid[208] is an example of interpretation of graphical images. In this effort, the researchers used samples from a controlled and uncontrolled corpus of digital photographs to examine angles of light projecting shadows in two-dimensional images of three dimensional objects. They found that, for the examples they used, when there was a single major source of light (i.e., the Sun), angles of shadows from the apex of the sources of those shadows to the corresponding points on the shadows varied by only about 5%.

Based on these results, they assert that photographs where shadows vary by 25-30% indicate that the photographs were composites rather than original images. Without going into details of the underlying issues, this sort of analysis, based on assumptions such as the single source of light, may be highly problematic. Conditions could readily be created in which the assumptions are not true but for which the technique would yield the same results. For example, if there is a prominent highly reflective structure out of image that reflects the sun on one part of the image but not another, it could dramatically change the apparent angles for one part of the image and not others. But in testifying in court, without the detailed knowledge of whether this is

208 Hany Farid, "Digital Image Forensics", National Academy of Sciences, Annual Meeting Symposium, Legal/Forensic Evidence and Its Scientific Basis, April 25, 2006.

the case or not, the analysis may be presented as if it was definitive when it is not. This is an example of interpretation in the form of extrapolation.

Over-interpretation of traces and going "a bridge too far"

Over-interpretation is common in the digital forensics arena, as can be seen by examining typical testimony and reports. For example, the following statements are from actual legal matters, and are made without sufficient basis (the specifics of the legal matters are not included).

> XXXX of the emails contain subject lines that claim the recipient has either been approved or pre-approved for a mortgage. These emails contain a link that offers to get brokers to call the recipient, after providing confidential information. These subject lines are therefore misleading or completely false and intended to get the recipient to open the emails, as the sender has no previous knowledge of the recipient.

In this case an "expert" in digital forensics who is not a linguist is giving opinions about the meaning of language, making assertions about intent, and asserting facts not in evidence about the asserted lack of previous knowledge by the sender.

> YY of the emails contain subject lines that claim that there is a pre-existing application on file. Since the emails are trying to get a recipient to fill out a form with their personal information that will be used to get refinance quotes, this cannot be a truthful statement.

In this case the expert is asserting that something is impossible when there simply aren't enough facts to prove any such thing, and again interpreting language without proper credentials.

> The “WWW” [information] was filled in on the exact same page to which over YYY of the emails directed the recipient. ... This ties all of those YYY plus emails directly to the [Defendant].

Again, this goes a "bridge too far" in many ways, not the least of which is the assumption that, because two emails have the same

URL contained within them, they were sent by the same party or for the same reason.

Limitations of tools and false depictions in trace interpretation

Another common problem in interpretation of traces is the erroneous assumption that the tools used to examine traces are perfect and that their users are also perfect. Neither of these are even remotely true, and these assumptions produce a wide range of difficulties.

One example of a widely used tool is the "EnCase" product, which has been roundly criticized in various forums for various reasons. Like Microsoft and IBM before them, being a market leader brings lots of criticisms, so these should be examined with an eye toward understanding and not assumed to be all true or justified. In one legal matter, an example that is likely present in many tools, illustrated the problem of tools interpreting traces. In this case, EnCase identified that a particular file was modified at a particular time in a particular time zone, and the time zone of the particular modification was indicated as in the middle of the Atlantic ocean. This particular date and time was critical to the case, and the time zone interpretation was critical to the specifics of who could have done what. One side indicated that the time zone was an anomaly that they did not know how to explain.

As it turned out, the time zone was the result of information given to EnCase by the examiner about the time zone of the computer (Eastern). But as it also turns out, the examination made this assumption during the summer, but the date was formed during the winter, while the date and time were kept internally within the document being examined in UTC. As a result, the daylight savings time assumption made by EnCase interpreted the UTC date as being an hour off from its normal value, which placed it in the middle of the Atlantic ocean.

> CAUTION: Do not over-interpret this example from EnCase. All presentation is a form of interpretation, and subject to similar sorts of errors.

This is a case of a tool over-interpreting data instead of simply presenting and analyzing it, and of the examiner being unable to

directly interpret the information because the tool presented it without the necessary trace details to allow the interpretation to be corrected by the examiner.

An independent review of some forensics tools

In one simple independent review of some tools,[209] a series of tests were performed to increase the confidence level of the tester in his use of forensic tools. This started as a quick test of basic functions, like correct image creation, hard drive wiping, and searching. He switched hard drive interfaces, computers, and other hardware, in an attempt to troubleshoot these problems, and paraphrased vendor responses to his concerns. To quote and paraphrase:

> "My background includes a little programming,... a lot of running new software, and some troubleshooting of operating system or hardware problems. I'm familiar with some of the literature on software engineering, and anticipated and actual bug rates. I don't expect software to be bug free - certainly I've never written or found any. But the lightweight testing I've done indicates that serious testing would probably find more serious flaws. It's also generally true that the more functions software has, the more bugs it has. I'm more concerned about flaws in core forensic functions...

Hardware 1

Brand A write blocker can't read a 2.4 GB IDE hard drive. Brand B write blocker reads it fine, as well as two different IDE interfaces. This isn't a "silent failure" problem, just an annoyance.

> Vendor response: First we've heard of it.

Software 1

1. Skips a sector while capturing an image, using its own write blocker. Potentially serious error, at least as far as examiner/software credibility.

> Vendor response: Yeah, we know about it.

---

209 C. Preston, correspondence submitted to the "CFTT" mailing list on or about Tue Feb 4, 2003 11:09 AM.

2. In wiping a couple different hard drives, fails to wipe what appear to be hundreds of sectors. Potential credibility problem for software/examiner.

Vendor response: known problem

3. In doing a search of a partition, the software reported 26 hits instead of 25, by adding the last hit twice. Not too serious, since the reported location was shown, and identical to hit 25. Except for credibility.

Not yet reported to vendor

Software 2

Capturing images that under some conditions are corrupted. Opening image exhausts all of RAM in a few minutes, application crashes.

Vendor response: We're really sorry. Ships new code right away. Problem still seems to be there. Ships more new code. Apologizes and still checking problem.

Software 3

Finds search term in the proper places until Unicode switch is used. Then finds nothing.

Vendor response: Asks for more details, offers to test

Software 4

Can't dump memory image of device. Installed application on two different computers, about 15 attempts, with suggestions from vendor tech support.

Vendor response: we're really sorry - we will have to locate one of those devices to check the software with

Software 5

Software mistakenly identified two text files as encrypted in the first run. Not serious if the classification is only wrong on a few files.

Not yet reported to vendor

Some software companies used to furnish a list of known anomalies with each software release. Would this be a good practice to follow, for forensic tools?

Problems with depictions from tools

In essence, all output from all tools used in digital forensics constitutes a form of interpretation. The tools present the underlying traces, which are bits, in a form associated with the media, in a human usable form, such as a display containing depictions of integers, hex codes, strings, listings, date and time indicators, directory structures, and so forth. The tools use the syntax and semantic elements they are designed around to interpret the traces and present those interpretations to the human users, who in turn interpret the depictions and write reports based on those interpretations. Each of the interpretations is potentially problematic unless the user is fully aware of the limitations and properties of their tools and understands what they do, how they do it, and how they fail under which circumstances.

Another example of this problem comes in the analysis of datagram sequences by packet analysis tools. In experiments on the use of deception for information protection, deceptive datagrams were introduced into a network that was subject to surveillance using network traffic surveillance tools.[210] The approach of the introduced traffic was to create deceptive datagrams that the tools used by the surveillance teams would present to their users as normal network traffic. The observers than examined the traffic, drew conclusions, and were fooled into making wrong assumptions and drawing wrong conclusions about the nature of the network. This ultimately led them to attack the wrong IP addresses, thus attempting to attack and/or use computers that did not exist (they made the same sorts of assumptions that others make about the association of IP addresses with computers).

Another example of a common misinterpretation by lay people, that should never fool a DFE examiner, is the depictions produced by the Internet site known as the WayBack Machine, which exists at

210 F. Cohen, et.al. "Leading Attackers Through Attack Graphs with Deceptions", IFIP-TC11, `Computers and Security', V22#5, July 2003, pp. 402-411(10).

www.archive.org.[211] The WayBack Machine depicts historical versions of Web sites, and as such, it is often used as a source for people trying to identify evidence of historical activities. While this site is useful for some level of entertainment and some level of historical understanding, from a standpoint of forensics, the depictions it provides are often very misleading. The basic problem is that the presentation mixes content from different dates together as if they were from the same date and time. Since content at URLs on Web sites may change with time, including, in particular, graphical images, the mixing of content from different time frames leads to depictions that place things in visual proximity even though they may never have actually coexisted on the original Web site.

This problem can be clearly seen in examples such as the depiction of the http://all.net/ Web site. In particular, by examining the depictions labeled as from 1998-01-20 at 02:13:37 and 1998-04-22 at 17:42:40 it can be clearly seen that they both have an image of the Naval Observatory clock. That clock showed 1:48:31 PDT for both time frames, and that time corresponds to neither of the identified times. In fact, the WayBack machine retrieved the graphical image only once, on 2003-08-10 at 18:48:31 GMT. The date and time indicated from the WayBack Machine URL corresponded to the time stamp of the Naval Observatory Clock and was from more than 5 years after the date and time of storage of the original content from all.net. This behavior was confirmed by the WayBack Machine's Web site which indicated that this is how the mechanisms worked. Only one image of the clock was taken, and that was taken at a different time than each of the other elements of the Web site. Since the same image names or filenames may be used for different content at different times, the mixing of the content together may result in complete fictions being portrayed to the user.

But as the operators of the WayBack machine learned of this problem, they then "fixed" it, and it now works differently than it did in 2008. As of this writing, the all.net Web site from 1998 had a date and time indicating whenever the viewer looks at the WayBack machine's depiction. The historical depictions from the WayBack

211 The "Wayback Machine" is located at http://www.archive.org/

machine that showed one sort of error up until 2008 now show a different sort of error. What error will they show tomorrow?

Such examples have happened in legal matters and the all.net Web site was used in reports in two such cases to show that the depictions of the WayBack machine were problematic in this way. In one such case, the Plaintiff sued the Defendant for running a different business using the computer facilities of a previous joint business that they both owned. The evidence was from the WayBack Machine, but the dates and times of the different parts of the composite image of the Web page portrayed, and that formed the "evidence" in the legal action, were from different dates and times. By using a timeline, it became clear that the images presented as from the new company were indeed taken after the new company formed and the old company no longer existed, while the portions of the depiction indicating the old company were from before the old company broke up and the new company was formed. In essence, a movie could be created that showed how the depicted image appears to have the two companies simultaneously depicted on the Web site, when in fact they could not be shown to have coexisted at any time whatsoever. This is a clear example of expert interpretation that flies in the face of the obvious interpretation that a lay person would and did produce from the same depiction. It exemplifies how computer-based tools can present false impressions and why experts are required to interpret such evidence.

Interpretation of missing traces

Specific examples of missing traces in time stamps and records in log files have been used as part of a methodology for interpretation of traces.[212] To the extent that assertions are made about things that are not present in traces and that "should" be present, these are interpretations.

Missing traces and records are critical to detection of intentional modification, and are well known and widely identified, but rarely with scientific basis. For example, authors may say that there should be a record of some sort and that its absence indicates

212 F. Cohen, “A Note on Detecting Tampering with Audit Trails”', 1995, available at http://all.net/books/audit/audmod.html

tampering. They may come up with casual theories and assert them as fact, and in some cases they may do this as part of expert witness testimony or in their written disclosures.

This is problematic, in that there are many possible causes of missing traces and records, such as, (1) the records were never created, (2) the processes normally creating them failed in some way, (3) they were overwritten by an anomaly in the operation of the systems or mechanisms, (4) they were present but not detected, (5) the hardware failed, (6) a programming error caused them to be overwritten, and so forth. Unless all such possibilities are accounted for, there may be many unidentified consistent causes.

Record retention is increasingly viewed as mandatory, and many legal cases have now been tried in which records were destroyed improperly and those who failed to retain them were punished in court rulings ranging from fines to adverse jury instructions.[213] Intentional destruction of records after a party is or should reasonably be aware that these records might be relevant to a pending legal action is treated as intentional, and may be subject to criminal sanctions such as obstruction of justice. Government record retention requirements such as the Presidential Records Act[214] and other similar legal provisions are designed to require that records be kept, and their destruction is also criminal. However, the interpretation by the DFE examiner in formal results should be limited to identifying the traces that indicate records as missing, including identification of traces that are normally produced and are not present, identification of traces that might indicate the presence of these traces, and so forth.

For example, in getting records that include the configuration files of a server, the configuration file meta-data may indicate that the configuration was unaltered for a period of several months and that the configuration as specified normally produces records indicative of each access, including particular data within those records. The absence of these records in traces or the unwillingness of the party

213 "The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production", September 2004 Public Comment Draft.

214 The Presidential Records Act (PRA) of 1978, 44 U.S.C. ß2201-2207.

to produce such records may then be interpreted by the examiner as an indication that either the software being operated was not operating normally, that the configuration files provided were not the ones being used, that the records were not created, retained, or produced, that they were deleted, or that some other circumstance has made them unavailable. Each of these possibilities can be considered by the examiner as sets of events, with the hypotheses relating to these events generating tests that are then run against the traces. One after another of these hypotheses may be eliminated, depending on the specific results.

In some cases, the absence of records may be treated as an indication of spoliation, based on the examiner's understanding of the situation and detailed analysis of different consistencies and inconsistencies. For example, when the events indicated by a party or their expert produce results that are inconsistent with the traces provided, it is clear that either the events or the traces are not what they are purported to be. In such a case, the examiner may reasonably state something to the effect that, "Assuming that the [state the relevant events] as indicated by [state who indicated these things] are true and correct, this analysis shows that the [detail the traces] are inconsistent with [other party's] [statements or whatever type the events are]. I conclude, therefore, that either there is an otherwise undetected flaw in the analysis presented here, the [other party's] [statement or whatever] are not true, or the [traces] are spoliated and cannot be relied upon." Various rewording may be used depending on the specifics of the matter at hand, and of course, if there are many examples of such inconsistencies, the case for spoliation gets better and better.

The use of redundancy to mitigate interpretation errors

Analogous to the discussion for analysis, redundancy in tool use and process is also a sound way to increase the confidence in tools and methods used for interpretation. Interpretation errors are far harder to mitigate, in the general sense, than analysis errors, because analysis ultimately goes to very hard to dispute factual statements about traces and already interpreted assertions about traces, while interpretation goes to the meaning of words and conclusions based on assumptions.

With redundancy comes increased assurance that interpretations are correct. For example, in a spoliation assertion, the use of redundant methods for analysis of traces decreases the potential for errors in the claim and increases the credibility of the results. Simply believing that the use of a computer program or other technical mechanism works is inadequate to interpretation. If results are to be credible in the face of challenges, something must be done to verify the results. Otherwise, the results should not be relied upon by the examiner in interpreting their meaning.

As in analysis, interpretation may use redundancy in the form of examining intermediate results for consistency, doing sanity checks such as that counts are reasonably related to sizes of traces or analysis results, removal of "found results" from a copy of the original traces and re-examination of the remaining traces, and other similar methods. Tool imperfections can also be mitigated by redundancy, as they are in analysis. And of course, the more separate and different the "redundant" version is from the original, the less likely there is to be a common mode failure.

Evaluating trace interpretation with information physics

When an interpretation is being undertaken, it may be valuable for the examiner to question the interpretation in light of information physics. As an example, in the case of a claim of spoliation, other explanations may be possible, and they should be considered in light of information physics as a sanity check on interpretation. Table 6.1 is an example of a spoliation case review based on select information physics results.

Digital World	The event in question
Finite time granularity (the clock)	Could the appearance of spoliation be the result of clock granularity limits?
Finite space granularity (the bit)	Could the appearance of spoliation be the result of space granularity limits?
Exact copies, original intact	Is the spoliation claim verifiable against the original traces provided?

Digital World	The event in question
Finite (fast) rate of movement	Is the spoliation claim related to short time frames, and if so, are there other possible causes or invalid assumptions?
An artifice created by people	Could it be that the people creating the systems and mechanisms were at fault rather than the other party?
Finite State Machines (FSMs)	Could the appearance of spoliation come from errors in the original automata?
Homing sequences may exist	Could homing sequences explain the apparent spoliation?
Forward time perfect prediction	Can the behavior asserted be shown in reconstruction?
Backward time non-unique	Are there other possible events that could have caused the same results without spoliation as the explanation?
Digital space converges in time	Are there multiple significantly different event sequences that might explain the apparent spoliation claim?
The results are always bits	Is there any other interpretation of the bits that might allow for consistency?
Results are always "Exact"	What should the traces be, and how substantive are the differences?

Digital World	The event in question
Time is a partial ordering	If time is an issue, are there other consistent orderings of events that include clock changes that could explain the traces?
Errors accumulate	Is it possible that what is seen as spoliation is simply error propagation, perhaps across a discontinuity?
Representation limits accuracy	Could what appears to be spoliation merely be a lack of accuracy?
Precision may exceed accuracy	Is the precision of the analysis too high for the traces?
Forgery can be perfect	Are there any indications of forgery? If not, is the spoliation perhaps unintentional?
DFE is almost always latent	Are the tools used accurately presenting and properly interpreting the traces?
DFE is circumstantial	What circumstances could be consistent with the traces?
DFE is hearsay	Could it be that the underlying mechanisms that created the traces are simply unreliable?
DFE cannot place a person at a place at a time	Could a third party be responsible for inconsistencies in the traces? Could an attack on one of the computers have caused the inconsistencies?

Digital World	The event in question
DFE can show consistency or inconsistency only	Could other traces or events show that the current traces and events are consistent? What traces or events are they, and where could they be found?
Probability is dubious	What certainty level can be associated with the claim of spoliation and how would that certainty be defined?
Content has information density	Is the spoliation associated with a density or similar measure? Is this a reliable indicator of spoliation?
Content density variable	Is some other indicator present that might indicate another sort of content is present?
Digital signatures, fingerprints, etc. generated from content	Is there any indicator produced by the content that can be used to test the spoliation claim?
Content meaning is dictated by context	Is there another interpretation of the syntax or an environmental condition that could explain the traces?
Context tends to be global and dramatically changes meaning	Is there any missing context, and if so, how could it effect the claim of spoliation?
FSMs come to a conclusion	Were all of the tests definitive, and what did they indicate?
Cognitive limits from program	Does the trace indicate what really happened, or is it only an indicator of what might have happened?

Digital World	The event in question
Hardware fault models from computer engineering	Could a fault have produced the trace or made it inaccurate, and in what ways?
Time and space tradeoffs known	Is the asserted spoliation possible within the available times it is asserted to have happened?
Near perfect virtualization and simulation possible	Is it certain that the traces reflect actual events, or could the traces have been produced in a virtual environment and be accurate within that context?
Many nearly or equivalent FSMs	Are other FSMs present that could produce the same results?
Undecidable problems	Could undetected consistent events produce the traces?
Computational complexity limits computations	What analyses that could have rehabilitated the traces were not performed and why?
Consistency is guaranteed	How were the traces produced? If this can be explained, could it rehabilitate the traces?
Completeness is guaranteed	What was the coverage of the analysis against what models, and what does this coverage indicate about what was and was not tested?
Time limits on achievable results	Is the asserted spoliation achievable in the available time with the available expertise?

Digital World	The event in question
Complexity-based designs	Is there some digital fingerprint or other similar mechanism that can be used to demonstrate that the traces are forged?
Fault tolerance by design	What fault models can explain traces inconsistency, and what are the reliability figures for those classes of faults?
Accidental assumption violations	What sorts of accidents could have caused these traces?
Intentional assumption violations	What sorts of intentional acts could have caused the traces?
Discontinuous space	Are there any discontinuities in space that could have caused the apparent spoliation?
Discontinuous time	Are there any discontinuities in time that could have caused the apparent spoliation?
Minor differences amplified near discontinuities	Are there amplification effects that go to the spoliation issue?
Major differences suppressed away from discontinuities	Are there suppression effects that go to the spoliation issue?
Identical use of an interface may produce different results	Is it possible that the interface caused apparent spoliation?
Ordering may be reversed	Are traces not in proper order and is this from spoliation?
Value sorts may be reversed	Are there values that are meaningful that might have been misread at the interface?
Actuate-sensor loop errors	Are there any actuator - sensor loops that might have produced errors in the trace?

Digital World	The event in question
Sensors/ actuators limited in physical properties	Are there physical issues associated with the process that cause errors in the trace?

Table 6.1 - Applying information physics to ask questions about spoliation

Given this range of possible issues and the typical quantity of interpretations involved in a legal matter, it may be infeasible to fully examine each of these issues for every interpretation. But it is typically pretty easy to dismiss many of these issues with only minimal attention to the assumptions stated in the interpretation given.

Interpretation of events

The context of the case and of the traces, along with all of the other events, such as; declarations of the parties, statements made under oath, charges, claims, countercharges, counterclaims, legal rulings, and other such things, provide events. They are typically in the form of sentences in human languages, and these expressions must be interpreted in order to be used for analysis and in interpretation of analytical and other results.

The interpretation of words and implications in events

Typically, the examiner receives statements in words, such as "This computer was used to print these checks". This might be backed up with a variety of other statements attempting to relate analytical results to the event, such as:

> "On July 4th at or about 12:38 PM, the Word program sent a document titled "JoJo.doc" to the printer identified as Laser 13, as shown in [whatever the filename of the printer log is in this case] at lines [relevant line numbers in the trace].

In examining this event, a claim made by a party, the examiner must somehow interpret it in a manner that allows the claim to be tested against the traces. In such a case, the examiner might identify that the traces should have particular patterns that can be found by searches, and based on these interpretations of the events, may extract traces and search those traces for indicators of the asserted event.

The mere appearance of a time stamp in a file, while it may be consistent with the event asserted, may not be the whole story. Indeed, if this was all of the available evidence, it would leave a great many questions about the issue. For example, an interpretation may assert, without limit, that:

- The dates and times indicated are accurate.
- The Word program was operating on that computer at that time.
- The Word program was able to send documents to printers at that time.
- The programs and libraries supporting the actions of Word were present, properly configured, and active at that time.
- The word program had access to the document titled "JoJo.doc" at that time.
- The printer identified as Laser 13 was accessible by Word at that time.
- The log file indeed shows what is claimed.
- The fonts used in the printout were available to Word on that computer at that time.
- The spacing, kerning, and justification were available from Word in the version present on that computer at that time.

The examiner that wants to assure that the events are consistent with the traces should also go through the more thorough process of examining the traces for internal consistency, starting in this case, at the level of the file system; the claims about the behaviors of the various programs involved and their presence on the system; the accuracy of the clock-related information that produced these records; the printer log file and its proper sequencing and other related characteristics; and so forth.

The examiner may also go a step further by recognizing that, if this printer is a laser printer, it might be one of a type that produces tracers and those tracers may be checked for consistency by

comparing the tracers produced on the original evidence to the date and time stamps, printer type, make, model, and so forth.[215]

All of these steps are interpretations that the examiner makes of the events, and in many cases, there are many events. This act of interpreting events is not computerized at this time, and there appears to be no breakthrough on the horizon that will change this. It is, in many ways, like the word problems commonly used in engineering and mathematics classes, where the student must learn to interpret the problem and put it into mathematical terms in order to find an equation to solve the problem. The interpretation of events and the identification and creation of tests that might detect consistencies between events and traces is, in large part, what the skilled DFE examiner does, and this is where their expertise comes into play.

It is critical to be clear that there is no systematic way to translate events into hypotheses or into specific traces that should or should not be present. As discussed earlier for analysis, interpretation cannot be done perfectly or completely in any realistic situation. Like any other scientific process, hypotheses are created by the examiner, tested against the available events and traces using a limited set of consistency checks, and the results are reported out. There may always be another consistency check that was not done or that may be thought of by someone else at some later time, and failure of that test might show some set of hypotheses to be inconsistent with each other or the traces. But there is no realizable number of tests that can done to comprehensively cover all of the possible hypotheses that can be made.

Event interpretation in light of information physics

Typically, different examiners will come up with different lists of things to test in order to determine the consistency of events with traces. As we know, there is no known limit to the number of such tests that may be revealing because the total number of traces is so large, the interpretation of language is not precise, and the nature of DFE is not well understood. There is some hope of being

215 D. Schoen, "Investigating Machine Identification Code Technology in Color Laser Printers", 2005, The Electronic Frontier Foundation, available at: http://www.eff.org/wp/investigating-machine-identification-code-technology-color-laser-printers

thorough in the effort, in the sense of covering a set of known possibilities. As a baseline, the different properties of information physics can be examined with respect to each of the events and all of the traces to identify testing methods that may indicate whether the information physics is violated, and thus detect inconsistencies. For example, taking the statement regarding printing above, we can go through each of the stated and implied events and identify examples of where information physics might speak to the events and traces of import to this matter. Table 6.2 starts down this path.

Digital World	The event in question
Finite time granularity (the clock)	Is the event characterized within the clock granularity?
Finite space granularity (the bit)	Is the event characterized in terms that can be translated into bit settings?
Exact copies, original intact	Is there an original that the final output can be compared to for consistency checking?
Finite (fast) rate of movement	Is the time reflected on the printer possible given the times of other related events?
Finite State Machines (FSMs)	Are all of the relevant FSMs present required to do the asserted actions?
Homing sequences may exist	Are the available traces consistent with the possible states required to produce the event?
Forward time perfect prediction	Can the behavior asserted be shown in reconstruction?
Backward time non-unique	Are there other possible events that could have caused the same results identified without the asserted event being true?

Digital World	The event in question
Digital space converges in time	Is there lost information in the traces from convergence that leaves uncertainty about the history asserted?
The results are always bits	What portions of the event cannot be tested through the bits?
Results are always "Exact"	Are there any indications that the traces are not exactly as they should be? What should they be?
Time is a partial ordering	Do the traces demonstrate consistent orderings with the events?
Errors accumulate	Are there any errors that would be detectable in the traces that are probative with respect to the events?
Representation limits accuracy	Are the results accurate to the expected level, and if not, what are the sources of inaccuracies?
Precision may exceed accuracy	Is the precision of the event within the accuracy of the mechanisms it speaks to?
Forgery can be perfect	Are there any indications of forgery? If not, what might a forger have to have done to produce results this good?
DFE is almost always latent	Are the tools used accurately presenting the traces and properly interpreting them?

Digital World	The event in question
DFE is trace but not transfer	Is there any assertion relating to the event that is based on transfer?
DFE is circumstantial	What remains circumstantial about the event and traces and what circumstances could be consistent with the traces and not with the event?
DFE is hearsay	What is the basis for asserting that the traces are normal business records, what other related records are there, how accurate are they, and does this meet the legal standard that applies?
DFE cannot place a person at a place at a time	Who commanded the event to take place? Was it the user present, someone remote, another program, or can it not be determined?
DFE can show consistency or inconsistency only	Are all examined traces and events consistent? Could other traces or events show that the current traces and events are not consistent? What traces or events are they, and where can they be found?
Probability is dubious	What certainty level can be associated with the event?
Content has information density	Is the density of the content consistent with the nature of the environment?

Digital World	The event in question
Content density variable	Is some other indicator present that might indicate another sort of content is present?
Digital signatures, fingerprints, etc. generated from content	Is there any indicator produced or missing that can be used to test the event?
Content meaning is dictated by context	Is the trace consistent with context syntax and semantics? Is there another interpretation?
Context tends to be global and dramatically changes meaning	Is there any missing context, and if so, how could it effect the meaning of the records?
FSMs come to a conclusion	Were all of the tests definitive, and what did they indicate?
Cognitive limits from program	Does the record indicate what really happened, or is it only an indicator of what might have happened? Could the record be present even though the file was never actually printed?
Hardware fault models from computer engineering	Could a fault have produced this record or made it inaccurate, and in what way?
Near perfect virtualization and simulation possible	Is it certain the records reflect actual events in the asserted system, or could the records have been produced in a virtual environment or simulator?
Many nearly or equivalent FSMs	Could a nearly or equivalent FSM have produced the traces?
Undecidable problems	Could undetected inconsistent events produce the traces?

Digital World	The event in question
Computational complexity limits computations	What interpretations were not performed and why?
Consistency is guaranteed	Are there any inconsistencies, and if so, how were they produced?
Completeness is guaranteed	What was the coverage of the analysis against what models, and what does this coverage indicate about what was not tested?
Time limits on achievable results	How much time was available and used, and how much more time would it take to do other tests that might refute the ones used as the basis for the conclusions?
Fault tolerance by design	What fault models can explain how the traces could be inconsistent with events, and what are the reliability figures associated with those classes of faults?
Accidental assumption violations	What sorts of accidents could have caused these traces if the event did not happen?
Intentional assumption violations	What sorts of intentional acts could have caused these traces if the event did not happen?
Discontinuous space	Are there any discontinuities in space that could be in place that might refute the event?
Discontinuous time	Are there any discontinuities in time that could be in place that might refute the event?

Digital World	The event in question
Minor differences amplified near discontinuities	Are there amplification effects that go to the issues?
Major differences suppressed away from discontinuities	Are there suppression effects that go to the issues?
Identical use of an interface may produce different results	Is it possible that the interface caused actions or interpretations that are identified rather than knowing acts of people?
Ordering may be reversed	Is it possible that the events are not in the proper order?
Value sorts may be reversed	Are there values that are meaningful that might have been misread at the interface?
Actuate-sensor loop errors	Are there any actuator - sensor loops that might have produced errors in the traces?
Sensors/ actuators limited in physical properties	Is the printed output consistent with the printer claimed to have produced it?

Table 6.2 - Information physics questions about the example event

Given this range of possible issues and the number of events and traces involved in a typical legal matter, a truly thorough review of even the limited things identified in information physics seems infeasible within the constraints on resources normally involved in the legal system.

Some limited metrics for consistency interpretation

One analysis of error uncertainty in DFE provides a proposed scale fore qualifying the utility of DFE.[216] This scale identifies 6 levels of certainty as summarized in Table 6.3:

216 E. Casey, "Error, Uncertainty, and Loss in Digital Evidence", International Journal of Digital Evidence Summer 2002, Volume 1, Issue 2

L	*Description / Indicators*	*Qualification*
0	Evidence contradicts known facts	Erroneous Incorrect
1	Evidence is highly questionable	Highly Uncertain
2	One source of evidence not tamper resistant.	Somewhat Uncertain
3	Source(s) of evidence are tamper resistant but not sufficient for firm conclusion or unexplained inconsistencies present.	Possible
4	Evidence is tamper resistant or redundant independent sources agree.	Probable
5	Evidence is redundant from tamper resistant sources. Only small uncertainties exist (e.g., temporal error, data loss).	Almost certain
6	Evidence is tamper proof and unquestionable.	Certain

Table 6.3 - Casey's metrics for consistency certainty

Casey asserts out that level 6 is unattainable today and provides examples of how combinations of evidence with different certainty levels might be combined to produce a resulting overall level of certainty. However, no systematic method for combining items from these levels is provided, no calculation method is shown, and the decision as to what level each item belongs in is an interpretation exercise.

In context of the approach in this book, these levels might reasonably correspond to type C consistency with level 0 corresponding to -1 and level 6 corresponding to +1.

Resource limits and interpretation - the schedule

It is the interpretation of the examiner in the context of the schedule and associated resources that dictates which questions will be asked and what tests will be performed to get at the answers to these questions. When the attorney asks the seemingly simple question: "What else might be relevant?", the answer may be very substantial indeed, and "I don't know." might be the best answer.

The schedule sequence (S) forms an *a priori* unknown sequence of situations that the examiner must deal with over the course of a legal case. From a practical standpoint, this often means that:

- Very little time is available to make preliminary determinations that later have to be backed up with more detailed analysis and interpretation, or found to be in error.
- The available budget to support the forensic effort may limit the sorts of analysis and interpretation undertaken. This then means that the potential interpretations are limited and that in-depth exploration will be focussed on the key issues in the legal matter.
- Over the course of the case, additional traces may become available up until some date at which discovery ends. Up until that point in time, special care must be taken in interpreting events and traces, in that future developments may alter the interpretation; some of the events might be removed from consideration; and some of the traces or events may be corrected or better traces found.
- After discovery closes, expert reports and responses are due, and the examiner may have to answer challenges to their interpretation and test the analysis and interpretations of other examiners. These are done in relatively short time frames, typically on the order of 30 days. Greater care in preparing written materials and making statements at depositions and hearings will make this process far easier and assure that the most accurate information is provided.
- New interpretation methods may become available through research and development or unavailable after legal rulings.
- Examiners sometimes create their own tools, at least in the form of combining the functions of other existing tools. Testing these tools then becomes part of the resources used in meeting the schedule.

The schedule also has impacts on the processes associated with analysis and interpretation. For example, different sorts of tests and validations may be done on systems and software, but these take considerable amounts of time, depending on the specifics involved.

The schedule may make certain types of procedures too time consuming or expensive, and the result may be a lower standard of care.

The prioritization of processes, analysis, and interpretation is a task of the examiner in working with the legal team, and interpretation may limit the utility of their work.

Interpretation in statements and reports

Every time the examiner makes a statement or writes a report relating to DFE, there is an inherent interpretation in that the writing or statement translates from the mathematics of analysis into the human language, with all of its imperfections and potential interpretations by others. The selection of words is interpretive unless explicitly guided by a defined methodology, and such a methodology does not exist in common use in DFE examination today. But even if a common language did exist, this would not completely ameliorate the issue. Even the precise wording advised in areas like statistics, does not change the fact that there is interpretation in the decision of what reliability assumptions are used for the different statements and why it is that those levels constitute a meaningful breakpoint for making a statement of one sort rather than another. Statistical conclusions in legal matters are not ultimately interpreted by experts in statistics, and the statements of DFE examiners in legal matters are not ultimately interpreted by DFE experts. The trier of fact likely does not understand the subtleties of interpretation and takes expert statements at face value.

As in the interpretation of traces, the interpretation of events can go too far or have errors that create the opportunity for various legal challenges. Here is an example from a real case:

> An IP address is a unique numeric address used by computers in the Internet. An IP address looks like a series of numbers, each in the range of 0-255, separated by periods (e.g., 121.56.97.178). Every computer attached to the Internet must be assigned an IP address so that the Internet traffic sent from and directed to that computer may be properly directed from its source to its destination.

This extract is fairly standard language in affidavits used by law enforcement in trying to get search warrants and for other similar purposes. The problem is that it is not factually accurate, even though it is a pretty reasonable approximation of some aspects of how IP addresses and some parts of the Internet work. For example; an IP address is not numeric (it is a set of octets); not unique (many computers for example have the IP address 10.0.0.1); it only looks like that when depicted in that way and can look like other things; many computers that are attached to the Internet have no such address, including a wide range of switches and routers that are address free; and it ignores IPV6, which does not have these sorts of IP addresses.

In some cases, complaints may be unclear as to what the issue is in the case. For example, a law may say that "it is illegal for a person to send [content] to another person", and someone who received such content may make a charge against someone who sent that content. But suppose the claim reads something like this: "A sent the [content] and C received it." This does not mean that "A sent the [content] to C". Depending on how the examiner interprets the claim, they may indeed show both that "A sent the [content]" and that "C received the [content]", but if they interpret this to mean that "A sent the [content] to C", this is a mistake. Suppose that "A sent the [content] to B" and "B in turn sent the [content] to C". In this case, C could charge B, and B could charge A, but C could not charge A. The examiner should make it clear in their interpretation that, even though the facts are consistent with the traces, this does not imply that "A sent the [content] to C".

There are any number of other similar logical fallacies that may be encountered in events, and the examiner who is trying to evaluate traces in light of events should be cognizant of these issues and identify them when they are present. It is the job of the examiner to interpret events and traces together and to be careful and picky in their interpretations.

Notions of "similarity" and quantification

One area where examiners may interpret more liberally than in other areas is in the evaluations of things that are "similar" in some ways. This is an area where interpretation is very touchy and

problematic. The examiner who wants to assure that their answer is right will take steps to move away from similarity issues or clearly define what similarity means and how it is characterized in terms of differences. This goes to the fundamental issues of how expert evidence is to be evaluated by the courts.

In essence, when an examiner makes a statement about things being similar or very much alike, or any other such interpretation of analytical results, it is then incumbent upon them to bring sufficient clarity to the use of these terms to allow the trier of fact to be clear about what that means in terms of the legal matter and how reliable that similarity is in terms of being probative in the case.

So close and yet so far?

As an example, suppose the examiner finds many messages, files, audio recordings, or other content that seems to be nearly identical, and yet are not strictly identical. A sound file might sound like the same song when played, but not be a bit-for-bit match to the content of another sound file. Messages may have identical bodies and similar headers, except for some portion of the header fields being different. Files may print out and look nearly the same after printing, but may contain different representations or character codes. Documents or directory structures might have similar elements, systems may have nearly identical files, user accounts may have similar passwords, computers may have almost identical IP addresses, and so forth. But all of these similarities may be deceptive in the extent to which they imply a relationship.

It is up to the examiner to be careful in making claims of similarity. It is critical that the examiner apply their expertise and not apply expertise that they do not have in making determinations about similarity in light of scientific principles associated with the specialized areas of the interpretation. For example, for someone who knows a great deal about how Internet email mechanisms work; examining headers of a collection of emails may lead to the interpretation that emails with identical "Message-ID:" field content except for apparent monotonically increasing sequence numbering at the end of the "Message-ID:" fields, received from apparently different senders from different IP addresses using different mechanisms, is inconsistent with the normal operation of the mail

transfer agents at issue. And yet, there are conditions under which this can occur in the normal operation of electronic mail systems.

Even those who believe that they are experts in email operations in the Internet may disagree over the interpretation of the traces and events for this case, and disagree further on the potential causes of such apparent inconsistency. Resolving this issue may require more than experts and their interpretations of traces and events. The examiner wishing to interpret this seeming inconsistency should be careful to recognize that the similarity issues here must be thought through and resolved before drawing any definitive conclusions or creating interpretations that may be incorrect.

Substitutions and similar comparison mechanisms

Interpretation can also be aided in many cases by the creation of derivative traces. For example, by replacing syntax elements by other syntax elements, it often becomes far easier to detect differences and identify consistencies and inconsistencies. One simple example is the use of hyphens within text documents at the ends of lines to allow syllables to appear on different lines in the printed text. The inclusion of a hyphen in this manner will make text searches for strings go awry because they will almost certainly miss strings that are altered in this or any of the other ways that similar things get altered in use. Similarity can be detected by substituting a hyphen followed by a newline and any indentation by nothing, leaving the original word intact for the purpose of searches, and allowing for the interpretation of such a syntax element as the original word.

Another example of substitution is to replace all syntax elements not of import to the matter at hand with a standard character, such as an asterisk (*), and then search the remaining content thus disregarding the intervening syntax without losing track of the fact of its presence. Any number of other sorts of replacements may also be used, but care must be taken in interpreting the results. These sorts of substitutions may, depending on the specifics, lead to false results. In these cases, it will be most prudent to perform the analytical process but return to the original traces for the interpretation to verify that the processing has not altered the manner in which the content should be interpreted.

A fairly clear example of the misuse of this kind of substitution technique is the substitution of URLs in Web searches or field specifiers in database searches. While as an investigative tool it might be reasonable to try many different combinations of inputs to different mechanisms to see what they generate as outputs, for forensic purposes it is problematic. The fact that a database, Web site, or other mechanism that is not fully understood by the examiner, responds in some way to something similar to what occurs in original traces, may be almost meaningless. For example; if a database when searched for records containing a 127 in a particular field (A), yields a "not found" result, the fact that the use of the 127 in a different field (B) or the use of a 126 in field A produces a valid record, is not a reasonable basis for interpretations such as: (1) there once was a record containing a 127 in field A, (2) a record in other fields will be similar to the records produced by other searches with similar values, or (3) almost anything else the examiner might want to assert. Unless there is some far more definitive information about the underlying mechanisms and content, this sort of interpretation is pure guesswork. It is not consistent with the standard of scientific rigor that would be reasonably accepted in a legal setting.

Measurements of similarity and caution in their use

In order for similarity measures to be reasonably applied, there needs to be some meaningful metric that can be reasonably applied according to a scientific methodology that produces testable results demonstrating that the similarity is probative with regard to the matters at hand. Finding such a metric and validating it is not a trivial matter, but it is necessary in order to make meaningful interpretations of similarity in cases such as those identified above.

Automatic content inspection methods

Methods used in intrusion detection systems (IDSs) and defenses against undesired content (UC) provide some useful insight into the problems faced in trying to match similar things. Both IDSs and UC systems have hard-to-define rates of false positives and false negatives. Each of them fails to recognize large quantities (even in cases where they are small percentages) of what they are

supposed to be trying to detect, and each incorrectly detects large quantities (even if they are small percentages) of things they are not supposed to detect.

There are fundamental reasons for these problems,[217] and these reasons have not changed over a substantial period of time. The reasons include, without limit; the undecidability of some of the underlying detection problems; the potentially large quantity of data that may have to be examined in order to make a definitive determination when one can be made; the long time frames over which detection may have to operate; the trivial manner in which intentional evasion may be done in most cases; and the difficulty in specifying what the similarity measure is supposed to detect and not detect and with what precision.

While reducing false positives and false negatives is certainly worthwhile for such systems, reducing one tends to increase the other. When there are large volumes involved, even small rates of false positives and negatives yield large absolute numbers of wrong answers. At the end of the day, results from all such mechanisms must be examined by the examiner for similarity, and the examiner's interpretation forms the basis for selecting out the "real" matches. This puts the last step of the method into the realm of opinion, and eliminates the quantitative metrics that may be applied to the automated results.

The number of human interventions in the process also translates into examination workload. Typically, as the workload increases, the methods become less desirable, so the mechanisms, which are generally tunable as to the threshold below which inexact matches are tolerated, are tuned to provide enough matches to fill the available examination resources. Further metrics may be used to try to do a more rational job of tuning, but such tuning is not generic and must be done for each sample set, assuming that the sample sets are not otherwise identical. Metrics like Shannon's information content[218] have been used to compare content density to language

217 F. Cohen, "Intrusion Detection and Response", National Info-Sec Technical Baseline, Lawrence Livermore National Laboratory and Sandia National Laboratories, December, 1996. See: http://all.net/journal/ntb/ids.html

218 C. Shannon, A Mathematical Theory of Communications, Bell Systems Technical Journal. 3, no. 27, (July 1948).

content density characteristics and detect things like cryptography and steganography. But again, these mechanisms are, in most cases, problematic for the same sorts of reasons as the IDS and UC methods. These methods, like the IDS and UC methods, may be helpful in an investigative process used by the examiner to explore the traces and events, but they are not typically as useful in interpretation.

Bloom filters and similar methods

A recent application of Bloom filters[219] for similarity analysis has been applied in presumptive testing. In this approach, an analysis that cannot be readily tracked to a specific basis is used to indicate candidates for similarity and the examiner then does a detailed analysis to try to identify meaningful interpretation. A Bloom filter uses an array of *n* bits (called keys) to store the results of *k* independent hash functions, each with an output range of 1.. *n* inclusive. Each hash function is run on each of two input sequences to produce sets of 1s in the bit array. Any bit set for one input sequence and not for the other indicates a non-identical pair, but identical outputs does not prove identical inputs. As *k* and *n* grow, the likelihood of false positives drops, and the time to do the test grows. Thus time is traded for accuracy. By creating multiple subsets of content (called chunks) from input sequences and applying Bloom filters, the percentage of identical hashes can be used to quantify similarity with respect to those chunks.[220] This then leaves the problem of selecting chunks, hash functions, and values for *n* and *k*, and implementing a mechanism. Automatically selecting chunks was done by experimentally identifying regions of the input sequence with Shannon's information content determined to differentiate similar from dissimilar inputs. To quote “It is difficult to formally analyze and to prove that this context-based approach to hashing yields reliable results. However, experimental results show that, for realistic targets, the basic idea is sound and works well.”[221] Thus “presumptive” positives are produced that have to be

219 Burton Bloom, “Space/time trade-offs in hash coding with allowable errors”, Communications of ACM, pages 13(7):422-426, July 1970.

220 Vassil Roussev, Golden G. Richard III, and Lodovico Marziale. “Multi-resolution similarity hashing.” Digital Investigation, 4(S):pp. S105–S113, 2007.

221 Vassil Roussev, Golden G. Richard III, and Lodovico Marziale. “Multi-resolution similarity hashing.” Digital Investigation, 4(S):pp. S105–S113, 2007.

verified by the examiner. The results don't provide a detailed basis for an assertion of similarity, and higher results often do not indicate of more similar content in any humanly discernible way.

Other similar dubious interpretations

As many of these examples of interpretation have shown, there are many pitfalls for the examiner who seeks to make a case by finding ways to justify a position or is less than careful in the extrapolation or inference process. Interpretation can easily stretch from the credible and reasonable into the ridiculous and embarrassing. It is a slippery slope that has to be carefully considered at every step before being trusted for use in a legal matter.

There are widely published classes of logical fallacies that are commonly found in human endeavors, and they can be accidentally or intentionally applied to draw all sorts of conclusions. Thomas Gilovich provides in-depth analysis of human reasoning fallibility by presenting evidence from psychological studies that demonstrate a number of human reasoning mechanisms resulting in erroneous conclusions.[222]

The mechanisms identified in Table 6.4 should be avoided by the examiner in interpretation as elsewhere, and the examiner should explicitly attempt to notice these mechanisms if and when they occur, both to avoid them and to identify them in analysis and interpretation by others:

Fallacy Mechanism	*Example*
Effects should resemble their causes	
- instances should resemble their categories	Similar looking traces are more likely to be from the same mechanism than less similar looking ones. (This sounds good, but what is the scientific basis for its use in interpreting traces?)

222 Thomas Gilovich, "How We Know What Isn't So: The fallibility of human reason in everyday life", Free Press, NY, 1991.

Fallacy Mechanism	*Example*
- like resembles like	Log files containing the "SMTP" string are produced by mail transfer agents. (Does this mean that a trace of a failed user login where the user typed "SMTP" as the user ID was produced by an MTA?)
- tendency toward oversimplification	If it looks like a Web "get" request, it must be a Web "get" request. (Does this mean that a description of the protocol with an example is an actual request?)
- Occam's Razor	The trace of the "get" request is consistent with the presence of a Web server on the computer. (Is the simplest answer necessarily the right one? What if no Web server could be found? Does this mean a server was there before?)
- black and white	The trace is from a Web browser. (Consistency of traces with the use of a Web browser does not mean that a Web browser produced it.)
- rule of 3s	There are three reasons for [you name it]. (Are there really only three reasons, or is this all they could think of?)
The misperception of random events	
- the clustering illusion	Traces showed that 65% of all the access occurred during high tide. (Is there really a relation between tides and access?)
- over-application of representativeness	There were three traces of activity 1, and in each trace, it was followed within 30 seconds by a trace of activity 2. (How significant is this as a correlation, given that there are only 3 samples?)
- misperceptions of random dispersions	There are too many traces within one hour for this to be accidental. (What is the scientific basis for this interpretation?)

Fallacy Mechanism	*Example*
- the creation of casual theories	This date and time stamp are wrong because [theory]. (Is this the only possible reason?)
- the regression fallacy	More traces of faults were generated in the week this software was installed than were seen in the previous month, so the software was clearly creating problems. (How do we know that the faults weren't just fluctuating normally?)
Misinterpretation of incomplete or unrepresentative data	
- the excessive impact of confirmatory information	We saw nine examples of the same thing, so it must be true, and the anomalies we saw were because of [some previously unanticipated reason]. (Why don't the anomalies count as refutations?)
- the tendency to seek confirmatory data	In looking for out-of-order traces, unclear orderings were treated as out-of-order because we didn't want to miss any of the potential evidence. (Possible in-order traces were treated as out-of-order so the theory that was being put forth could be bolstered by more instances?)
- the problem of hidden or absent data	27% of claims about messages from Joe were inconsistent. (Does the examiner have all of the messages ever sent from Joe?)
- self-fulfilling prophecies	We suspected the computer was used to print the checks, and when we looked, we found a form with sizes and shapes consistent with printing them. (If we look hard enough, we may find such a form on almost any computer.)

Fallacy Mechanism	*Example*
The biased evaluation of ambiguous and inconsistent data	
- ambiguous information is interpreted in context	The trace indicates that the printer was used on the same day as each check was cashed. (This can be over-interpreted as either supporting or refuting the use of the printer for printing the checks. Was it also used on other days?)
- unambiguous data is shaded	The time stamp in the trace of printing on Monday indicated a time after the check was cashed that day, but that was the day after we changed back to standard time. (The change to standard time moves the clock forward, so a failure to change it on the computer would not make the time on the computer later than bank time.)
- multiple endpoints	The identical Message-ID on two emails indicates a carbon copy. (But it might also indicate a duplicate record or forgery or any number of other things.)
- confirmations and non-confirmations	Out of 1.5 million files, only 15 had create times after modification times, but this is within the normal range of out-of-order files on a computer. (That doesn't mean that in this case it was "normal".)
- focused and unfocused expectations	We looked for traces indicating that Joe was logged in and the program was running and found 3 of them. (And how many cases when others were logged in and the program was running, when Joe wasn't logged in and the program was running, etc.?)
Outcome asymmetries and one-sided events	

Fallacy Mechanism	*Example*
- hedonic asymmetries	We found two nearly identical images but the background was changed from blue in one to green in the other. (The 16 bit difference in blue and green backgrounds may be treated as more important than a 5000 byte difference between two images with blue backgrounds. And how do we know blue was changed to green instead of green changed to blue?)
- pattern asymmetries	At or about 1:11 AM and 2:22 PM each day, the file was modified. (The analyst may remember or focus on clock settings when they are 1:11 or 2:22 better than when they are 1:08 or 2:19.)
- definitional asymmetries	Performance got slower and slower until they stopped attacking the system. (Since "attacking the system" is not well defined, it is always able to be true, since we can call the time that performance improved the time when the attack stopped.)
- base rate departures	"These messages slowed our server" is supported by the time to deliver those messages. (But it ignores other events in the server that might have produced the slowdowns in delivery of the messages.)
Motivational determinants of belief	
- empirical support for the wish to believe	The hourly traces of data entry counts confirm the claim that data was entered at the same rate at all times of interest. (The traces appear to confirm the event when the examiner is trying to find consistency, but an hourly statistic may be less revealing than a shorter time framed one.)

Fallacy Mechanism	*Example*
- mechanisms of self-serving beliefs	Examination of the traces found three instances where User 15 was logged in and the suspicious activities took place. (But what about all of the instances where one or the other were present alone? If you want to believe it you ask "Can I believe it?" while if you don't want to believe it you ask "Must I believe it?". Scientific interpretation gives more power to refutation than confirmation.)
- optimistic self-assessment	"It's a slam dunk." (We all think we are right most of the time.)
The biasing effect of second hand information	
- sharpening and leveling	"The traces are consistent in all but 17 examples." (as opposed to "The traces are inconsistent in 17 instances")
- the corrupting effect of increasingly indirect evidence	The game of 'telephone' is a great example - hearsay evidence is excluded for this reason.
- telling a good story	They clearly intended to [whatever]... (DFE examiners should not opine on psychological issues like "intent" unless they are also psychologists.)
- distortions in the name of informativeness	Joe is a hacker and the payroll system was hacked. (The terms "hacker" and "hacked" are misnomers, imprecise, and problematic - but they sound good.)
- distortions in the name of entertainment	"There is one example of..." becomes "I have seen these sorts of things before" and is interpreted more like "... is common".

Fallacy Mechanism	*Example*
- distortions in the name of self interest	Clearly [followed by almost anything]. (As a rule of thumb, any definitive statement about the real world based on DFE is problematic and not likely "clear".)
- distortions due to plausibility	More than 70% of all network traffic in the Internet is "spam". (It sounds plausible, but what could be the real basis for this? How is "spam" defined? How was this measured and why do we believe it reflects the real totality of traffic in the Internet?)
Exaggerated impressions of social support	
- social projection and the false consensus effect	Most experts would agree that ... (How exactly would this agreement be gained?)
- inadequate feedback from others.	"This method was published at a DFE conference in a paper titled "..." (The quality and nature of the referee process for conferences is highly non-uniform and the mere presentation at a conference does not constitute peer review.)

Table 6.4 - Logical fallacies common in humans applied to interpretation

These mechanisms are detailed and supported by substantial evidence. Most of them are believed to be common to most individuals in all human societies. Counsel in a legal matter, someone from the other side, or others working on the examiner's team, may come up with cleaver approaches or interpretations that, while they seem sensible, are in fact not valid at all. It is up to the examiner to use appropriate care in interpretation to avoid these pitfalls, and to present credible expert interpretation to their clients and the triers of fact.

Interpretation and similarity

A great deal of research has been done in searching large collections of content for strings or other similar expressions, including searches for n-tuples and proximity. For example, searching large portions of the Internet for small sequences of words is now commonly done by millions of people on a daily basis using services like Google and other similar sorts of search engines. These issues have been studied throughout the history of computing, and have been the subject of well known works.[223] [224] These techniques include providing metrics for similarity and sorting of search results based on those metrics for the purposes of presentation. Many digital forensics tools provide capabilities for searching for regular expressions, which identify class sets of syntactic sequences that could be identified as "similar" in that they are part of the same specified set. The same is true of search engines built into or added onto operating systems, such as the Apple Spotlight mechanism.

There are also techniques such as stylometrics,[225] based on writing or coding styles and the use of pre-existing code collections from known locations such as Web sites, news groups, books, and less widely available sources, graphic design style, vocabulary, sentence structure, word usage, etc. Pedersen has been a leader in this sort of research,[226] and there are many examples of methods

223 D. Knuth, "The Art of Computer Programming, Volume 3: Sorting and Searching", ISBN 0-201-03803-X, Addison Wesley, 1973. [This book summarizes a wide array of research in computer science, and is the third in a series of books highly regarded in summarizing computer science research.]

224 S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine", Computer Networks and ISDN Systems, Volume 30, Issues 1-7, April 1998, Pages 107-117. [Defines the architecture for the initial Google search engine and identifies properties and features of search engines since that time.] (http://infolab.stanford.edu/~backrub/google.html)

225 C. Chaski, "Who's At The Keyboard? Authorship Attribution in DigiEvidence Investigations", International Journal of Digital Evidence, V4#1, 2005. [This paper summarizes results in attribution relating individuals to actions based on behavioral characteristics and identifies the extent to which the US legal system to date has accepted such results in court cases.]

226 T. Pedersen, "Computational Approaches to Measuring the Similarity of Short Contexts: A Review of Applications and Methods", Journal of Intelligent Systems (Special Issue : Recent Advances in Knowledge-Based Systems and Their Applications), 17(1-3), 37-50, 2008. http://www.d.umn.edu/~tpederse/

he has applied to analyze linguistic patterns for applications ranging from disambiguation of word sense in human sentences to detecting plagiarism.

While these approaches appear to be good ideas, and some of them have been significantly explored over the years,[227,228] from a DFE standpoint, there is little definitive information that can be used to associate reliability information with them. These methods do not scale well and are, at best, presumptive. Some such methods have been admitted in US courts for limited purposes,[229] but as a class, they have not been well tested or survived meaningful challenge, and research indicates that they readily susceptible to deception.[230] These methods generally attempt to attribute metrics to known individualized samples and then detect the presence of similar metric values within collections of traces associated with relatively small number (on the order of a few hundred) of known individuals to identify traces potentially attributable to the identified individual. They usually rely on N-grams of some sort, where the selection of the symbol set and sequencing criteria are defined by a syntax within a linguistic syntax POset. This is sometimes couched in a limited form in terms of "near", "next-to", "before", "with", and other similar search modifiers using distance metrics. This type of approach returns to the problems of identifying potential causal mechanisms, the E→C assumption, and the problems of symbol set and syntax selection that lead to factorial time and space.

[This paper summarizes results in examining n-tuples and proximity measurements for natural language processing to determine similarity and attribute word sequences to authorship.]

227 M. Corney, "Analysing E-mail Text Authorship for Forensic Purposes", Masters Thesis, Queensland University of Technology, March, 2003 [This thesis examines using a variety of classifiers with output fed into a Support Vector Machine (SVM). The approach compares a specific email to an SVM model built from a corpus of emails with known provenance e.g. given 20 emails from each of A, B and C, compare a new email to identify the author.]

228 F. Iqbal, H. Binsalleeh, B. C.M. Fung, and M. Debbabi , “Mining writeprints from anonymous e-mails for forensic investigation“, Digital Investigation, 2010.

229 C. Chaski, "Who’s At The Keyboard? Authorship Attribution in DigiEvidence Investigations", International Journal of Digital Evidence, V4#1, 2005.

230 P. Juola and D. Vescovi, “Stylometric Approaches to Author Obfuscation: An Empirical Study”, IFIP TC11.9 Digital Forensics Conference, Orlando, FL 2011-01-31-2011-02-02

Similarity of groups of traces performed so as to identify what groupings of traces are present and measure the extent to which they are similar, has not apparently been explored in any significant way. While there are some mathematical problems related to cliques[231] and a wide range of other similar sorts of things, these have failed to address the challenges of digital forensics in terms of the need to identify groups of traces with content containing similar characteristics and features. In forensics cases, while searches for known or suspected content are often used, it is also quite common to have a corpus of traces (e.g., files, messages, database entries, or other structured or unstructured content) for which identifying similar or related groups of traces becomes a key issue in addressing legal issues, particularly the issues related to attribution.

A typical example is a case in which attribution of "forged" USENET postings to real authors, systems, or mechanisms is of import.[232] Other examples include, without limit, cases where similarity of authorship, sourcing, or delivery mechanisms is probative; cases where evolved versions of similar coded content, such evolutionary viruses or copyright infringement with non-identical content; cases involving alterations, such as image files created or edited with the same version of the same software package; cases involving metadata where files with similar metadata may be related; cases involving log entries where similar sequences of events may be found; and cases in which common authorship based on writing "style", word usage, typing errors, spelling errors, grammatical constructs, etc. are to be identified.

Another general area of potential applicability is in intrusion, anomaly, or behavioral detection and analysis; where groupings based on identified characteristics may be used to associated large numbers of items of interest with each other.

231 I. Bomze, M. Budinich, P. Pardalos, and M. Pelillo, "The Maximum Clique Problem", The Handbook of Combinatorial Optimization (http://reference.kfupm.edu.sa/content/m/a/the_maximum_clique_problem__265525.pdf). [This book chapter summarizes mathematical results related to the identification of cliques.]

232 Susan Polgar vs. United States of America Chess Federation et. al. Case # 5-08CV0169-C.

In such cases, algorithmic complexity of $|T|^2$, where T is the set of traces being considered and |T| is the number of traces being compared to each other, seems almost inevitable, simply because, at least notionally, each trace $t \in T$ must be compared to each other trace in some way. While this produces feasible solutions for cases with relatively small |T|, as |T| grows, $|T|^2$ grows far faster. In cases involving 10^6 traces, $|T|^2=10^{12}$, which is near or over the edge of available time and space in typical legal matters. Cases in the legal system have already involved examination of almost 10^6 traces, in the form of electronic mail messages, far greater volumes are common in network analysis, and a typical file system today has millions of files.

In attempting to provide metrics for similarity, various authors have created measures of different sorts, typically for pairs of items being matched one against the other, or for matching of a regular expression or similar description against a set of items, such as for search engines. For example, two items that are identical in every way could reasonably be called 100% similar, and if a string "12374382302398" is found within one of the items being searched, the item could be reasonably reported as 100% similar based on the criterion that it contains that sequence. However, for less trivial situations, similarity metrics may be problematic. Similarity metrics for groups were only developed in 2009 as a method to identify groups of items with defined maximum sets of common factors. This is called the greatest common factor (GCF) method.[233]

Using the GCF method, some obvious metrics appear. Statements such as "Similar in n factors", or "Have x common factors", may be applied to the groups created. For example, a set of messages may be identified as having 18 common factors in their headers. But while this method may turn normal metrics (the factors) into interval metrics, this ability to count does not meaningfully address the question of "Counting against what?".

While there may be a temptation to make statements such as this group is similar in 18/360 factors, the notion that they are 5% similar is essentially meaningless. It is relatively easy to create

233 F. Cohen, "Identifying and Attributing Similar Traces with Greatest Common Factor Analysis", Pending publication.

factors to drive those percentages up or down for any given group. For example, examiners can remove factors not part of the desired results and produce "100%" similarity results, once they know what to choose as factors. Without a standardized base, ratio metrics, such as percentages, are meaningless. Furthermore, the question of how to weight factors and/or groups of factors depends on the particulars of the application. For example, to interpret the attribution of telephone calls to callers, and with groups that have identical calling phone numbers, this would seem in most cases, to be more important than the fact that there are large groups of calls of particular durations. Thus, for the attribution of calls to callers, phone numbers may be weighted more highly than call duration. However, if text messages are being sought, then call duration of relatively short time scales will almost certainly be a far better indicator than the phone number from which the call was made.

As a result, it is likely the best approach at this time to define metrics such that the features of interest to the application are weighted prior to analysis, ratio metrics are eschewed in favor of interval metrics, and to the extent that ordinal metrics may be applied to factors, a POset that may be developed to rank results with identical interval values. Without further information about specifics of the situation, the use of approaches like weighted sums may be more misleading than helpful.[234]

In order to be useful in a legal situation, results of analysis should also be meaningfully presented in terms of the specific basis for claims and presentable so as to demonstrate those bases.

Some of the techniques used for similarity analysis, like support vector machines (SVMs), and so-called “predictive coding” methods use training to form parameters that, like Bloom filters, don't directly relate to human-meaningful sequences.[235] Resulting similarity metrics have not been demonstrated meaningful from a standpoint of identifying mechanisms for cause and effect. In addition, these

234 Nominal metrics consists only of lists of things with no basis for formal comparison. Ordinal metrics implies a partial ordering. Interval metrics implies the ability to count things, but not against any standard. Ratio metrics implies the ability to add, subtract, compare, and normalize to a common zero value.

235 M. Corney, "Analysing E-mail Text Authorship for Forensic Purposes", Masters Thesis, Queensland University of Technology, March, 2003

approaches have only been applied with statistically meaningful results in cases where known good sample traces (i.e., no forgery or subversion) of all parties from a small group (on the order of a few score) are available to test against a suspect trace that is also assumed to not be intentionally altered, forged, or subverted.

Such mechanisms also do not produce independently verifiable results, in the sense that, without access to the mechanism used to derive the results, they cannot be tested. Thus we must trust the mechanism to properly perform its function, which leads back to the problems of proof of program correctness, specification issues, and all of the other challenges of trusted computing. It would be preferable to have mechanisms that allow an independent third party to examine the source content and the claimed results with minimal tools (e.g., a viewer of some sort) to confirm, on a case-by-case basis, that specific results are as claimed.

For example, output of the "diff" program that shows differences between two sequences can be verified by making the indicated changes to one sequence and verifying that it produces the other sequence. This can be done manually for small numbers of differences or with automation (e.g., using a text editor and macro processor) for larger sequences with more differences. In either case, reproduction and verification of results can be done with independent software on a different system and with limited effort.

Finally, and fundamentally, the Effect→Cause assumption inherent in the use of similarity is a fallacy in that, as information physics shows, there are clearly cases when different causes produce similar effects. As a result, similarity of effect does not necessarily imply similarity of cause. In addition, the lack of all traces being identical could be considered a refutation if this is inconsistent with the hypothesis of common cause underlying the notion of similarity.

These two criteria; (1) independent verifiability of results with limited effort, and (2) presentation of results in a manner that allows them to be verified and demonstrated to independent reviewers with limited technical understanding (i.e., the triers of fact) are vital to success of the claim of similarity in the legal system.

The Abstraction-Filtration-Comparison method

A method for evaluating similarity of intellectual property, and more particularly computer software, that has been accepted in many US courts since 1992 [236] for non-literal copyright violation is called the Abstraction-Filtration-Comparison method.[237] This method consists of three steps:

- **Abstraction** identifies the author's “expression” as opposed to their “idea”. It is undertaken by repeatedly abstracting the elements of the software at court-defined levels of; (1) individual instructions, (2) groups of instructions organized into a "hierarchy of modules", (3) the functions of the lowest-level modules, (4) the functions of the higher-level modules, and (5) the "ultimate function" of the code. Lower levels are considered more “expression” while higher levels are considered more “ideas”. Copyright law protect expressions but not ideas, and thus lower level correspondence tends to be more indicative of derivative work in violation of copyright.
- **Filtration** is used to remove elements of the abstraction that are not protectable under applicable statutes. Three specific things excluded in copyright are elements part of the public domain, externally imposed elements (e.g., the interface between a program and hardware requiring the use of a specific protocol), and elements dictated by efficiency (e.g., a sorting routine with $n*\log_n$ execution time).
- **Comparison** is then used by the trier of fact, as opposed to the expert, to determine whether and to what extent they are similar, and by implication, in violation of copyright. At the multiple levels of abstraction, it is hoped that similarities in expression will be readily understood and found, and the difference between expression and idea differentiated.

236 Mark A. Lemley, Peter S. Menell, Robert P. Merges, and Pamela Samuelson, Software and Internet Law (3d ed. 2006). ISBN 978-0-7355-5864-9

237 Computer Associates International, Inc. v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992).

Making assumptions (hypotheses) in interpretation

Assumptions, also known as hypotheses in the larger context, as potentially risky as they may be, are often required in order to make progress. Assumptions allow inconsistencies and/or consistencies to be found, and to the extent that the assumptions are confirmed by the traces and/or events, they are, at least weakly, supportable. If the assumptions are refuted by the traces, they are likely wrong, and should be abandoned with notation that they are refuted and why. In essence, the process of science is largely about making such assumptions and testing them to determine which ones are consistent and inconsistent with the facts. Given the notion that assumptions will be made and tested, which assumptions should be made by the examiner, and why?

Assumptions provided to the examiner

In many cases, an examiner has assumptions provided to them by legal counsel for one side or the other. For example, even though the events are not all available to the examiner, legal counsel may, in the context of a discussion related to schedule, reveal that some set of assumptions may be made in the case. These sorts of assumptions can be treated as events by the examiner, and should be documented in the examination report or otherwise clearly stated and tracked as assumptions in moving forward. In some cases, such assumptions are inconsistent with the traces and events, and such inconsistencies should be clearly identified.

Making assumptions "favorable" to the other side

While special masters may be assigned to cases in some instances, the predominant mode of examination is working for one party or another. In such instances, the examiner is working to show the truth from the point of view of one or more of the parties and is not working as a "fair broker". This does not grant license to lie or ignore the facts, but it does shade the manner in which activities are performed. The activity of the examiner almost surely involves some claims of another party that are being refuted and other claims of the client that are being demonstrated. In refuting a claim from another party or confirming a claim of the client, it is often effective to assume that some of the other party's asserted events are true and, based on those assumptions, refute the other

party's claims by showing inconsistencies. For example, if another party asserts that some set of records reflected in traces is true and accurate (event e_1), and makes some other statement about those records (e_2) that analysis shows is inconsistent with the traces, it is helpful to openly assume that e_1 is true. Indicate that this assumption favors the other party's asserted point of view, and cite e_1 as the basis to demonstrate the inconsistency of the traces with e_2. In doing this, care should be taken not to limit future interpretations so as to retain that assumption. It might be that e_1 was not correct.

Making assumptions based on trace analysis

Trace analysis often leads to reasoned interpretations, particularly in the area of trace type. For example, if a trace is asserted to be a forensic image of a disk drive, the first 512 bytes (an area typically used as the partition table) is formatted as a known type of partition table and indicates a type for a partition, and the identified area of the disk is of the proper format and syntax to be that sort of partition, it is often interpreted (and thus assumed) that the defined area of the image reflects a partition of the identified type.

If that assumption is made, further steps may be taken to analyze traces. Further assumptions about system initialization and operation may be made that indicate that the system containing this trace was running a particular operating system and version, that particular programs were in use, and so forth. Again, trace consistency tends to confirm these hypotheses and inconsistencies tend to refute them.

Of course all of these interpretations that are then used as a basis for further examination are hypotheses in the context of their use for ongoing analysis and interpretation. The chain of interpretations and analyses may become quite long, and the longer it is, the more potentially damaging a flaw in the early assumptions may be. For this reason, greater care should be taken in foundational hypotheses and there is all the more reason and potential benefit for those hypotheses to be demonstrated false by other parties. The hypotheses form a partially ordered set of interdependencies on which interpretations depend. To make the case less subject to challenges, increased redundancy in the dependencies helps, and

in particular, maximizing the weighted minimum cut of the partially ordered set tends to increase the workload of those trying to refute the interpretations.

As a simple example of a refutation of the entire chain that may develop from an initial typing, it may be argued that the disk image at issue was not from a disk that was used to bootstrap the computer, and that therefore all of the assumptions about the operating environment are faulty. The traces may be consistent with all of the activities associated with a bootstrap of the computer from that disk were undertaken, and this would appear to be inconsistent with the assertion that the disk was not in fact used to bootstrap the system. A counterclaim that the image is of a virtual computer and not the real computer might be made, and so forth. This is the nature of the interpretation process.

Making assumptions based on consistent events and traces

The addition of events to traces as a basis for making assumptions can be a great help in resolving such differences as those that come from the use of traces alone. For example, in the discussion of how the traces came to be and what they reflect, events may include statements by the representatives of a business that a computer in question was standardly configured to start up in a particular manner and that the normal business records reflected in the traces were generally relied upon for the purpose of tracking the use of computers within the business. This then helps to bolster the typing results from analysis and defeat counterclaims as inconsistent with the traces, the analysis results from those traces, and the events reflected in statements about the way these computers were normally used.

There is a cumulative effect that goes to the weight of the evidence. Counterclaims without the traces to back them up are more and more difficult to use as the set of events adds to the set of traces and analysis results. The combination of these factors allows interpretation to be more definitive, but at this point in time, there is no metric that can be reasonably used to assert the degree to which such combined results are definitive. Rather, the examiner may assert that in order for the alternative interpretation to be the case, each of the relevant sets of consistent traces, events, and

analysis results must not be the case. The alternative claim, in order to have credibility, should show how some or all of the traces and events are consistent with the alternative interpretation(s), and explain any remaining inconsistencies.

Making inconsistent assumptions

Perhaps that last "(s)" went unexamined, but it is important to the issues in examination. There is no requirement for consistency in legal matters. For example, if one party is making a claim, the other party need not rely on a single counter to that claim, and in fact, the various counters to the claim may be inconsistent with each other. For example, if an individual is accused of theft, the defense counsel need not show that they did not commit the crime. It may be adequate to show any number of different possible events that are consistent with the traces but inconsistent with each other, partially consistent with each other or the traces, partially consistent with other events, and so forth. The problem of finding all interpretations that are consistent with traces is, in general, unsolvable, and claims that one or another interpretation is the unique and correct one are problematic at best. In most cases, some inconsistencies go unexplained, and inconsistencies in traces sometimes occur even in controls used for experimental validation.[238] This is a form of background noise in the measurement.

There is also no reason that the examiner has to make all of their assumptions consistent. In fact, there is often substantial utility in making different and inconsistent assumptions to allow results to be demonstrated and to allow those results to be compared. For example, suppose that one party (A) claims one set of events and the other party (B) claims a different set of events. A sometimes helpful approach to interpretation is to do analysis based on each of the inconsistent assumptions of A and B and see which of them produces more or fewer inconsistencies with the traces. Presenting these results directly with minimal added interpretation may be extremely probative, particularly if one set of events is highly consistent with the traces and the other is highly inconsistent.

238 S. Willassen, "Methods for Enhancement of Timestamp Evidence in Digital Investigations", Doctoral thesis for the degree philosophiae doctor, Trondheim, January 2008, Norwegian University of Science and Technology.

In many instances, events are incomplete in terms of producing testable hypotheses for the examiner to check for consistency with the traces. In these cases, the examiner may decide to try different hypotheses that essentially add events to the analysis process, keeping in mind that the various hypotheses are not favored over each other. In many cases, this approach shows that one set of hypotheses is highly favored over others in terms of consistency with the traces. Subject to declaring the hypothesis as an assumption or conclusion with a basis, this is a viable approach.

Of course these assumptions may reasonably be challenged, particularly if one of the parties knows that it is wrong and hasn't revealed this previously. This also plays into the issues of the schedule, because the results of such interpretation are often revealed after the discovery process has ended, precluding the introduction of new evidence, such as a new fact that would tend to refute the hypothesis. This then gets into the area of legal strategy. As an example of such a challenge, there may be traces that indicate a particular piece of software was used in a particular system, and the interpretation may proceed with that as a stated assumption. At a later date, the other side may assert that the interpretation and the results that stem from it are wrong.

Legal strategy in interpretation

While an examiner who is doing their job properly should not be biased or particularly interested in the outcome of a particular case other than professionally, the job of the examiner does often include providing advice and information of a technical as well as tactical nature to the legal counsel for the party the examiner works for. In many cases an examiner will never testify or even write a report that gets shown in a legal process. Rather, the examiner's effort may simply assist legal counsel in making strategic and tactical decisions about the case and/or provide insight into what is really happening. In such cases, the examiner may be called upon to give general opinions, provide options of possible interpretations and how these interpretations might be made by others, what the likely costs and sorts of outcomes of different procedures might be, and other similar sorts of things.

Complex interpretations with assumptions

Many interpretations that would be very difficult to claim without assumptions may be made far easier to assert with assumptions. As an approach, it is not uncommon to start with those assumptions and try to find ways to analyze the traces and events so as to determine the consistency of those assumptions with the traces. A typical example is the attempt to show that an individual was at a computer continuously during particular intervals of time.

As usual, this depends on a lot of things, including the standard of proof and the available traces and other events. Typically, the behavior of the system, assuming there are various reasonable assumptions made, may be used to show traces of activities performed. Those traces may identify that those activities took place over particular time frames and at particular rates, and these traces may be leveraged to make other statements about the possibilities that an individual could or could not have left the computer during that time frame. Such analysis and interpretation is very complex and will be subject to substantial challenges if the other side is competent and the issue is key to the case and disputed. But this does not make it hopeless.

For example, (1) there may be events that demonstrate that the individual of interest was the only individual present in the relevant facility over the time frame; (2) traces may give no indication of any system subversions and be consistent with proper and normal operation over the period of interest; (3) events may stipulate that the traces are legitimate, taken from the specific system at the times indicated, and that that system operated with a standard version of a particular operating environment; and (4) the quantity of content and usage patterns reflected in traces of various activities such as the number of characters entered into a document over a defined time frame and Internet access traces such as the Web browser cache, may show that every few seconds some user input activity was performed. With a few carefully chosen assumptions that are confirmed by analysis of traces, a reasonable interpretation of nearly continuous presence might be established, where the term continuous implies that there was no period of absence in excess of some maximum amount of time during the larger period of use at issue.

As an example of the sort of error that is common in interpretation of this sort, in answering a question about how to prove beyond a reasonable doubt that a particular individual was continuously present at a computer over a period of time, a licensed private investigator within a firm that is "An Information Technology Detective Agency", answered "If you know how to use EnCase timeline features you can start from there. It's simple but can be tedious. Once you have all system activity between a certain date and time, reconstruct it in the report. Verifying authentications during that time are key as well..." Of course this analysis is inadequate to show that there was ever any person at the computer at all at any time, and interpreting such results as indicating continuous physical presence of a particular individual is clearly problematic.

Interpretation relating to hidden content

Hidden content interpretation is particularly problematic in the sense that the examiner is making claims (1) that normal tools don't demonstrate and (2) that are based on assertions about activities using software to alter traces to make them different from what they are normally interpreted to be.

The first problem that comes to mind is whether the traces provided are valid at all and whether the way they are being interpreted is valid. After all, the claim is made that things are not as they appear to be. Maybe they aren't what they are being presented as. Is the examiner saying that this Word document that is readily readable and writable by the Word program is not a Word document? Are the other Word documents also not Word documents? What else is hidden in there? How do we know that there aren't other programs hidden inside the programs being used to do the analysis? How can we trust anything about these traces?

In some sense, the claim is that this picture, which you can all see, through some complicated process that the examiner will describe, produces this completely different text hidden in the picture. The obvious questions that come up are things like:

> Suppose I wrote a different program that read that file and interpreted in a different way - would that produce a different text?

> You claim that this bit here and that bit there and so forth produce the text "The money is under the floorboard", but if I look at this bit here and that bit there, I find the text "The air is dry", and depending on where I look in the file, I can find almost any text I want, isn't that true?
>
> You claim that you found a "key" that unlocked the content and provided some particular string that you claim to be what was hidden. But what other strings could be revealed with different keys, and why is it that you think this particular string is the one that was originally put in rather than one of the other ones?
>
> Is this like the sounds we hear when we play Beatles records backwards?
>
> How do we know it wasn't hidden there by someone else?
>
> How do we know that there isn't a program that hid the real program that hid the text in the file and that the program that the examiner claims hid the text wasn't actually just a fake put there to fool the examiner into claiming that the text was hidden?

As the level of recursive hiding increases, this becomes increasingly difficult to explain. There is little literature that explores the general question of whether multiple keys may result in different results or the probability that a different key would yield a different result, or other similar sorts of information that might inform these results. Unless the examiner is able to interpret the hidden content in these terms, it may be problematic to assert that the content revealed is probative at all, or more probative than prejudicial.

One of the ways hidden content found in traces may be asserted as more reliable is if other traces and events are consistent with this interpretation. For example, if a program associated with a hiding technique is present, other known programs using other techniques are not present, the hidden information is consistent with the use of that program, system logs indicate the use of this program, and the times associated with the hidden content are consistent with the times at which the program was run, this will tend to support the assertion that the program was used to hide the content.

Visualization in interpretation and analytical product

The use of visualization and analytical product represent both traces and derived information based on traces in different forms. While probative presentation may be the intent, presentations and analytical products may deceive rather than inform.

The fundamental challenge of presentation as a form of DFE interpretation is to find presentations that preserve and accurately depict the traces in forms that tend to reduce human cognitive errors and misinterpretations. Since essentially all traces of DFE are latent in that they are not directly observable by human senses, the tools that make them observable color the human cognitive processes. The qualified DFE examiner is, by knowledge, experience, training, education, and expertise, qualified to interpret the presentations of the traces in context. But this can only be so if the tools are well enough understood by the examiner to make sound judgements about and interpretations of what those tools present and how that presentation comes to be.

At the level of observing a 1 or a 0 as depicted on a display or a page, this is relatively easy, subject to some assumptions about the conflict between an "O" and a "0" (the proper use of "a" and "an" should indicate which is which in this context). But when presented with something even as seemingly simple as a display of a date, examiners may readily misinterpret and the presentation may not be definitive.

For example, what is the date depicted by the display "07/09/08"? Is it MM/DD/YY or DD/MM/YY or YY/MM/DD or what? Is it clearer if presented as "7/9/8"? How about "2007/09/08"? Or what if the header on the column in which it is presented indicates "YMD"? What if the header indicates "YMD" and some of the contents indicate "2/3/98"?

The problem here lies in two distinct areas. One area is simply that what we see may not be what we think we see. This is purely a human cognitive problem combined with a display approach that fails to adequately distinguish things that are readily distinguishable at the level of the underlying traces. The other problem is that, in the process of analysis and presentation, there is concealed interpretation.

What you see is not what is there

The former problem, that the display mechanisms in common use do not adequately depict their input to allow forensic examiners to properly interpret them, is only beginning to be addressed.[239]

The problem can be seen in simple form in this example in which two files (test1 and test2) are compared using the "diff" program that identifies minimum differences between two files:[240]

```
FF>diff test1 test2
1,4c1,4
< This is a test
< This is another test
< This is a different test
< This is still another test
---
> This is a test
> This is another test
> This is a different test
> This is still another test
```

It seems clear that, in examining this output, there are differences detected between the files test1 and test2, but the lines asserted to differ look exactly the same. In understanding this output, the problem lies in the fact that the depiction doesn't fully capture the result. One approach to addressing this is to use fonts that provide clarity around the output without destroying the ease of use or ability to read the result. This approach has yielded a method to resolve the issues through the use of a different font and presentation. The goal is that:

- Each symbol should be clearly different from other symbols
- Each symbol should be familiar, with minimal interpretation, so that it looks similar to what might normally appear.

239 F. Cohen, "Fonts For Forensics", IEEE SADFE (in conjunction with the IEEE Oakland Conference), 2010-05-19, Oakland, CA.

240 J. Hunt and M. McIlroy, "An Algorithm for Differential File Comparison". Computing Science Technical Report, Bell Laboratories 41. (see http://www.cs.dartmouth.edu/~doug/diff.ps) June 1976. [This paper defined the algorithms used in the Unix "diff" program and introduces the problems associated with identifying maximum matching sequences.]

- Each symbol must be depicted, so that a <space>, <tab>, <carriage-return>, <backspace>, <escape>, and other "non-printable" bit sequences can be clearly seen on printed pages.
- Each symbol should self-indicate the bit pattern that produced it so that it can be traced back to its original value.

Here is a partial output of the same "diff" command issued earlier when depicted using a forensic font™ :

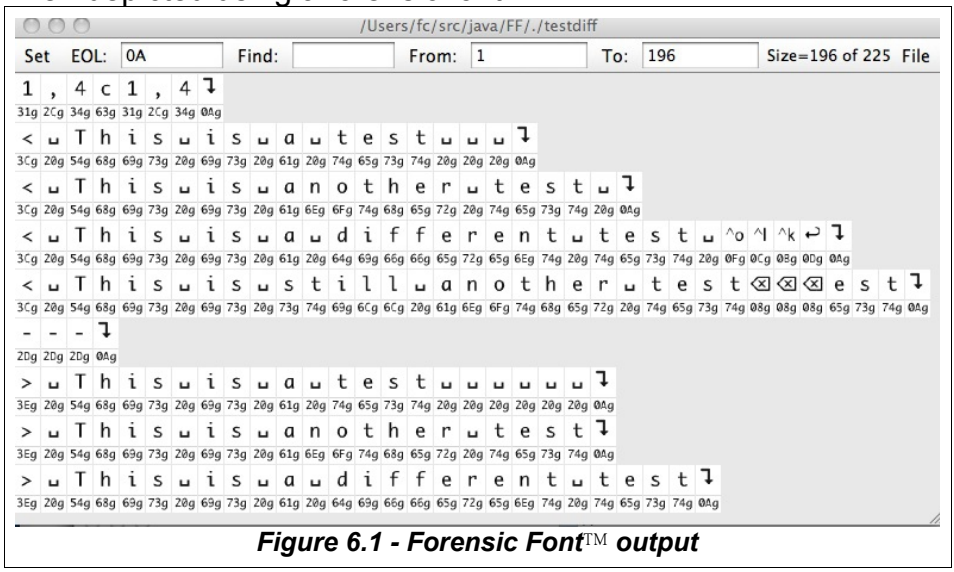

Figure 6.1 - Forensic Font™ output

This output shows that there are a variety of differences between the two files and depicts those differences to the examiner. In particular, there are trailing spaces, embedded backspaces and overprints, and other similar content contained within the different files so that they look the same even though they are in fact different.

This problem is pervasive. Even seemingly obvious things like empty lines and indentation are sometimes depicted in ways that make them impossible to detect from a printout. The examiner faced with such a document must explain this to the court, while the examiner who helps to prepare such a document has the opportunity to provide material in a manner that allows these differences to be readily resolved.

Concealed interpretation

A good example of concealed interpretation is date and time information that may involve time zone information that is not displayed or that is displayed in a context that shades the meaning. If the time zone is not depicted, there may be times from different time zones intermixed, creating ordering fallacies and similar problems to further analysis and interpretation.

Worse yet are cases where a time zone is depicted, but the time zone calculation is not correct for the context. For example, when indicating that Eastern Standard Time is in effect, a time offset from UTC as stored may result in an interpretation that is off by an hour if in fact Daylight Savings Time was in effect. It would be better from one perspective to keep the time in UTC for the purpose of analysis and use offsets for local times, but then matching up events such as statements that "At 2:30 PM, I ..." becomes an exercise in addition and subtraction.

Of course this is not just an issue with date and time information, or even with other sorts of orderings (or is that ordering of sorts?). When displaying the layout of items on a disk, the patterns may seem to indicate that the disk is relatively heavily used, but is it the display or the content that indicates this? It might be a few files that are dominating the disk space, created by someone who wanted it to look like they were doing a lot when they weren't. How does this reconcile with the file table usage patterns, the log entries of usage, and the delivered material provided by the individual that is supposed to be using the system?

Most modern tools largely fail to help the examiner address these sorts of questions and don't provide the means to ask these sorts of questions or get answers in a useful way.

To the extent that such tools are provided, the results are typically displayed in a relatively small number of different formats. These displays rarely provide the sort of complex cross-trace and cross-analysis examination or different sorts of visualizations required to move ahead in interpretation.

There are many human cognitive limitations associated with the interpretation of visual images,[241] [242] and the potential for self-deception is substantial. At the same time, visualization can lead to greater clarity in understanding large volumes of data, and some sort of visualization is clearly needed for interpretation. A basic question that has not been answered in the literature is where the tradeoff lies.

Interpretation errors and challenges

Clearly, information physics plays directly into the issues of interpretation, as well as into the notions of being thorough and comprehensive, and meeting resource constraints. Another way to look at this issue is through the lens of challenges in the sense discussed in Chapters 1 and 2. The interested reader should read more about these issues.[243]

The basic challenges are made and missed content, context, meaning, process, relationship, ordering, time, location, corroboration, and consistency faults. These can be accidental or intentional, and may produce false positives or negatives. In light of information physics and the interpretation issue, the following discussion may help to outline the relevant approaches to challenges, including the evaluation of faults in interpretation.

- **Content** is missed in essentially every legal matter, but the examiner starts with only the traces made available to them. Still, the examiner may be overwhelmed by volume and lack the resources to get to all of the traces available to them, and in every case is almost certain to only interpret a limited subset of all possible traces with a limited set of the available procedures. Challenges in the form of missed content are likely if the opposition has information that the examiner lacks or if the examiner doesn't have the same resources, tools, skills, or capabilities as the examiner on the other side. And even when these conditions are even, the different perspectives may yield different trace selections and

241 Donald D. Hoffman, "Visual Intelligence: How We Create What We See", Norton, 1998, NY.

242 Al Seckel, "The Art of Optical Illusions", Carlton Books, 2000.

243 Fred Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

different interpretations of events. The examiner should not make content, and to the extent that content is made, it is open to be challenged by a competent opponent.

- **Context** is particularly difficult to get right in interpretation unless the examiner has undertaken extensive efforts to reconcile every inconsistency identified in analysis. There is almost always some contextual information that the examiner doesn't know about, fails to thoroughly consider, or assumes to be the case when it is not in fact the case for the particular matter at hand. It is almost inevitable that experience taints the examiner to some extent, but whether these thoughts get translated into interpretations, is up to the examiner. The available schedule and the demands by the examiner for precision and testing in their work largely dictate the extent to which context is properly considered. At some point, all examiners must make assumptions, use their experience, and assume context in order to make progress. Whether this produces a false positive or negative is up to the opposition to figure out.
- **Meaning** in the context of the case is largely the result of interpretation. As such, interpretation produces the probative results. To the extent that results are not probative, the examiner should make this clear, and to the extent that meaning is implied, it is the job of the examiner to use interpretation to bring that meaning out with due care. The emphasis and manner of presentation may be said to alter the meaning, and the examiner has a responsibility to present interpretation that is true to the traces, the analysis, the events, and the scientific methodology in use. To the extent that this is not done, that information physics is violated, or that logical fallacies remain after interpretation, the examination may produce false positives and negatives in the meaning of the traces and events. This should be avoided to the extent possible and sought and refuted in examining interpretations of other parties.
- **Process** is used in all aspects of examination. In interpretation, process becomes a key component of what is stated by the examiner and how it is stated. To the extent

that the examiner fails to adequately identify the processes undertaken, the interpretation may be found to be without adequate basis. To the extent that these processes are incorrectly described, they may be found to be erroneous or improperly applied, or based on invalid methodologies. The examiner has a responsibility to be clear in the identification of methodologies and may be challenged if the methods used are not adequately vetted or if the processes that implement those methodologies are not adequately followed. This goes, among other things, to the validation and verification of tools and the manner in which they are used, the theoretical underpinnings of the field, and the soundness of the approaches taken.

- **Relationships** between and within traces and events are easy to miss, particularly since there are so many possible combinations. It is the expertise of the examiner that prunes the enormous number of possibilities down to a smaller subset that is manageable within the available schedule. It is easy to miss relationships that are subtle or indirect, and in almost every instance, such relationships are not examined to the full extent possible. In fact, no theoretical basis for complete pruning of the relationships has been put forth, and any such theory would likely be problematic. Missed relationships are a virtual certainty, but made relationships are an enormous problem. It is not unusual for examiners to make a relationship during the examination process and not have enough room in the schedule or awareness that the relationship was made to fully examine it. As a result, there are often challenges based on made relationships. To the extent that a relationship is key to a case, the examiner should take additional time and effort to assure that it is not made, but rather that it is verified to the extent feasible, and properly stated in reports and testimony.
- **Ordering** is a common examination issue and interpretation of ordering is key in many cases. Ordering is fundamental to causality and establishing causality is key to many issues in legal matters. Time is often used as an indicator of ordering and, if inconsistent with the ordering of traces, may be a

clear indicator of inconsistency. But the interpretation of ordering is particularly complex in light of the complexities associated with multiple processes and interferences among them in most modern information systems, the qualitative inadequacies of current systems and software, and the lack of common formats for auditing and recording of sequential activities within information technology. Particular attention should be paid to the difference between ordering and causality. Before ~~##~~ because.

- **Time** is a key element in interpretation in many cases, and time is particularly problematic in computers because of the wide range of possible errors in time sources, the high precision relative to accuracy of computer clocks, and the ease of manipulation of time mechanisms and traces. The interpretation of time and time-related information is often key to outcomes and the examiner must consider the range of possibilities and make determinations or assumptions in the analysis and interpretation of time-related issues. These determinations and assumptions should be clearly stated and, when alternative determinations or assumptions are possible, they may well be undertaken by the other side. Time-related misses or makes can produce dramatically inaccurate results, over- or under-interpretation of time-related data can produce astonishing results that are completely wrong, and the examiner must use proper care in this interpretation to avoid analysis failures.
- **Location** is often very hard to definitively identify from traces and events. To get a sense of this from non-digital evidence, a famous case involving the misidentification of an individual in the United States as being present at the Madrid train bombings in 2004[244] led to the dismantlement of fingerprint examination as a reliable forensic individualization method when it was ultimately shown that the individual identified was not in fact anywhere near the incident and that there

244 Statement of Glenn A. Fine, Inspector General, U.S. Department of Justice before the House Committee on the Judiciary Subcommittee on Crime, Terrorism, and Homeland Security concerning “Section 1001 of the USA Patriot Act” May 10, 2005, at: http://www.usdoj.gov/oig/testimony/0505b.htm

was no way in which he could have been the person who left the identified fingerprint. The same situation will likely apply to the excessive use of location-related indicators in DFE if it is allowed to be treated as definitive in the same way. Today, measuring the quality of location information is often problematic, and without metrics, conclusions are dubious.

- **Corroboration** is a key element in analysis as it reduces the potential sources of error and has the potential to produce independent verifications or refutations of hypotheses. While absence of evidence is not necessarily evidence of absence, if corroboration is treated as confirmation, its absence should be treated as refutation, albeit perhaps to a lesser extent. The examiner claiming that the presence of a trace would be an indicator of an act should also be prepared to admit that its absence is an indicator of the lack of that same act. Failure to produce corroborating evidence may legitimately be the basis for challenges based on the potential for incriminating or exculpatory evidence, and this is all the more viable if the available traces demonstrate inconsistencies that might or might not be clarified by the additional traces not available. It is a reasonable interpretation to assert that the lack of traces that are normally present to corroborate an event is consistent with the event not having taken place and inconsistent with traces indicating that the event did take place, just as it is reasonable to assert that the presence of traces that are normally generated from an event are consistent with the event having taken place.
- **Consistency** is the key defining principal of interpretation. Inconsistencies that are not reasonably interpreted have the potential to lead to challenges.

While these are not all of the possible challenges to interpretation, they are a good starting point for an approach to diligence in the interpretation of DFE by examiners, and should be considered by all parties in their efforts to seek just interpretations of DFE.

Questions

1. Given the key role of interpretation in linking traces to events and the inherently human nature of interpretation, how can the science of DFE examination provide a scientific basis for the interpretation process?
2. Occam's Razor apparently doesn't apply in the same way to DFE examination as it does to other scientific areas. When can this principle be applied in examination and when can it not be applied?
3. Given the notion that alternative explanations must be kept in mind and that events might be wrong, when can an event and a trace eliminate alternatives and to what level of certainty?
4. If there are no type C or type D inconsistencies found, does this mean that the only reasonable interpretation is that the events are as they are stated? If not, how could there be another interpretation that would be chosen over one that meets type C and type D consistency tests?
5. Given that added events can alter the view of type D consistency, are there conditions under which added events cannot so alter? What are those conditions?
6. Given the problems with statistics in the DFE examination process, are there cases when classical statistics are viable in interpretation? If so, what are those cases and why is this true?
7. Given that it is easy to go "a bridge too far" in interpretation, how can the careful use of language avoid doing so? What are some of the ways that interpretation can properly and improperly use language to indicate probative information about a case? How can the examiner tell when language goes too far?
8. In interpreting graphical images for shadows, identify three ways that two figures in the same picture that is not altered can have substantial differences in the angles of the shadows as they appear in the digital image.

9. How can the DFE examiner avoid the interpretation of traces by the tools they use? Given that this may sometimes be unavoidable, how can the examiner compensate for errors in such interpretations? Give an example using a tool that you use in examination to illustrate the problem and a way to address it.
10. Using the Wayback machines as a source, identify 10 sites for which the depictions presented by that source might fail to reflect the reality at any point in time, describe how these false depictions might mislead a trier of fact in a legal matter, and show a timeline that demonstrates to the trier of fact that the depiction is false for at least one example using the http://all.net/ site as the source for your demonstration.
11. How can the examiner identify missing traces, and how can the examiner be certain that these missing traces ever actually existed or might have existed? How could such an approach seem to be right when in fact it is wrong?
12. Taking one of the examples you generated from question 10, go through each of the items in Table 1.6 using the traces gained from the Wayback machine (not including the all.net example) and identify all relevant types of inconsistency with information physics that might apply. For each one that applies, identify how that information physics principle could cause the interpretation of the Wayback machine result to go awry.
13. Given an "on or about" statement in an event, how much of a time window should be provided before consistency is no longer justified?
14. Unless the DFE examiner is also a linguist, they presumably are not qualified to testify with regard to the meaning of words. How then can they justify interpreting the words associated with events when they claim inconsistency?
15. Given the enormous expansion of even a simple event into large numbers of potential consistency checks against traces, how can the examiner determine when to stop?

16. Describe how to answer each question in Table 6.2 in terms of what examination should be done, how the results will be used, and how the results will be interpreted.
17. Given that the schedule limits interpretation efforts, identify a process by which the examiner may determine what to do and what not to do by prioritizing the examination. Is this method usable in a real situation? How would it be applied?
18. Since modern digital rights management systems modify content to make each instance slightly different and to allow it to be tracked, authenticated, and to prevent illicit use, how can the examiner who does not have the necessary keys or mechanisms to bypass the rights management mechanisms compare two sound files, movies, or other managed content to determine whether or to what extent one is a copy of the other? How does the examiner interpret the similarities and differences meaningfully for the court?
19. Given that a change of even one bit can lead to a completely different interpretation of a sequence of bits, how can anything other than an exact match be considered indicative of similarity? What is the measure of similarity and how can it be systematically and scientifically applied?
20. Given the common logical fallacies identified in Table 6.4, should the examiner review each statement made by other examiners in cases they are involved in to identify if any of these fallacies are present? Will you do this for your own examination results? Do this for the last examination report you wrote or some other examination report you can find on the Internet or elsewhere. What did you find?
21. Given that progress often depends on making assumptions, how do assumptions get justified and tested, and if they turn out to be inconsistent with events and traces, what happens next?
22. Given the problems with visualization, how can the examiner ever be certain they are properly interpreting the results of examination? How certain can you really be?

7 Attribution

Correlation is not causality

In general, it is impossible to perfectly attribute digital traces and events to physical world events other than events within the finite state automata that form the digital world, and even then, attribution is limited by information physics. Nevertheless, the notion of causality is fundamental to all forensics, and the problem of attribution may be considered as identical for practical purposes to the problem of determining causality. This challenge has long been an issue in the digital forensics and secure computing arena, has been the subject of congressional testimony and[245] substantial research efforts,[246] and is of ongoing interest to the research community, government, military, and industry globally. The problem of causality in general and attribution in particular is profound at many levels.

The fundamental assumption of causality within the digital world is:

> **Traces come about by the execution of finite state automata that follow the physics of the digital world.**

The fundamental assumption of causality in the physical world is:

> **Effects come about by natural mechanisms that follow the physics of the physical world.**

Causality in the physical world stems from the nature of the universe and how it executes its underlying mechanisms. This is what the science of physics seeks to clarify. Causality in the digital world stems from the nature of finite state automata, which has well known mathematical characteristics within well known limits.

Like physics in the physical world, information physics in the digital world is a model. But the model in the digital world is precise while the models in the physical world are not, because the finest granularity of the physical world is not known, if it is finite at all. But the forensic challenges in both worlds stem not from the accuracy

245 F. Cohen, Feb 23, 2000 Written and Verbal Testimony before Congress, available at: http://all.net/journal/testimony.html

246 Survey / Analysis of Levels I, II, and III Attack Attribution Techniques", available at: http://www.cs3-inc.com/arda-survey.pdf

of the finest grain models. Rather, the worlds hold a common problem in that, even assuming that the models are perfect and at the maximum actual granularity, the complexity of the mechanisms are such that exhaustion of all possibilities is infeasible in almost all practical circumstances. This then implies that causality is not perfect in any but the most simple circumstances, in that the complete details of the execution of the underlying mechanism cannot be enumerated or reproduced with perfect accuracy.

The nature of statistics

Statistics is a mathematical field used to model and characterize systems and mechanisms that are not well enough understood or cannot be precisely modeled at reasonable cost. At a less granular level than the underlying physics, statistics provides the means to make certain assumptions and, based on those assumptions, attain defined levels of certainty about relationships between things.

In physics, we get statistical notions like objects that are mutually attracted by a phenomena called gravity that accelerates the objects toward each other at a rate that is proportional to the combined masses of the objects. This model, the one that is commonly used to explain the way apples fall from trees, usually has inherent assumptions about the apples being ideal spheres and the media through which the Earth and the apple approach each other being frictionless. More sophisticated models such as those taking into account the nonuniform surface of the objects, their shapes, and the friction of the environment they pass through, are also characterized with statistical approximations, and the models are made more and more complicated as the need for precision increases, up to some finite level of precision.

One of the underlying reasons that these sorts of models work and can reasonably and reliably characterize things such as the splattering of blood, the trajectory of a bullet, or the penetration of a spear into a body, is that minor deviations in the objects in question have little and generally compensatory effects on the outcomes. While wind alters trajectories of spears and blood droplets are altered by humidity, minor changes in wind and humidity do not largely or systematically alter the general pattern of blood splatter or the way a spear falls as it flies. This is a stability condition that, in

essence, asserts that minor changes in input produce minor changes in output. The traces will look very similar statistically.

While this model and statistics in general work well for gross behaviors and causal relationships in the physical world, they simply do not apply to the digital world in the same way. The reason for this is that the alteration of a single bit in an input to an FSM may have and commonly does have dramatic and non-localized effects on the output and state of that FSM. Outputs of digital mechanisms tend to be unstable with respect to single bit changes in inputs, so the underpinnings of statistical analysis do not hold true for digital systems in general. They may, however, hold true for specific digital mechanisms in specific cases. For example, in the reading of inputs from a digital scanner described in Chapter 4, the outputs are substantially different at the level of bits, but the images produced are all essentially indiscernible to the the human eye. Thus statistics about these pictures may be useful in characterizing them, even though at the highest granularity level, they differ. This is generally true for other sorts of unstructured data such as uncompressed sound, video, or image file content areas.

On the other hand, the reason that many security failures occur is that inputs to computer programs are interpreted differently than the designers intended them to be interpreted. A "buffer overrun" may alter executed code (the software-based FSM that is executing), unanticipated input sequences may drive programs into untested or under-tested operating modes, subtle interactions between sets of programs may occur with joint input sequences, environmental conditions within the digital world, like lack of disk or memory space or slowed processing may cause race conditions or unanticipated failure modes, and so forth.[247] Any or all of these may combine to produce states, outputs, and traces that are substantially different from those normally anticipated, and characteristics of these operating modes may differ greatly from those of more commonly observed conditions.

247 F. Cohen, C. Phillips, L. Swiler, T. Gaylor, P. Leary, F. Rupley, R. Isler, and E. Dart "A Preliminary Classification Scheme for Information System Threats, Attacks, and Defenses; A Cause and Effect Model; and Some Analysis Based on That Model", Encyclopedia of Computer Science, 1999.

The examiner therefore has to be very careful about the use of statistics in the attribution of digital traces to events, and in most cases, such statistics may be readily challenged based on the inherent instability of digital mechanisms.

If not statistics, how causality with complexity?

The problem remains of how to attribute effects to causes when the complexity of the automata are too high for precise models and statistical characteristics do not provide meaningful results. A comedian might say "Very carefully!", and so should a good DFE examiner. There are generally several different situations in which attribution can be reasonably well made, and in those cases, it can be made to a limited extent and based on assumptions and events.

FSM predictability

Some FSMs may be completely characterized. As examples, many hardware devices are defined, simulated, and implemented using hardware description languages. These descriptions provide the possibility of achieving tests with 100% coverage against certain known fault models. Given an input sequence and initial state, outputs can be definitively determined, subject to the hardware operating as designed. But great care should be taken in making such predictions without such a firm basis. For example, software operation in a complex environment has very large numbers of potential interactions with the environment.

Simulation approaches

Under somewhat harsher assumptions, that an exact copy of the initial state and input sequences with timing are available, it might be theoretically possible to simulate, emulate, or rerun the sequence of events that actually occurred. But this sort of reconstruction is problematic in terms of providing precise causality, because of the divergence properties of time and space associated with information physics around discontinuities. More discussion of reconstruction will be provided in the next chapter.

Complexity arguments and cryptographic mechanisms

In some cases, reasonable complexity arguments may be made based on computational complexity of forgery of keys, but such claims require assumptions or events asserting such things as:

- controls over and unavailability of keys, mechanisms that use the keys, and systems that control the mechanisms;
- a secure key management, creation, and distribution infrastructure;
- proper identification of individuals and their association with keys, systems, and mechanisms;
- the presence of specific individuals at specific places at specific times and their performance of specific tasks;
- the effectiveness of protection of systems and mechanisms used throughout the process;
- the purity of traces; and
- the low probability of certain things occurring within specific time frames in specific environments.

When many or most of these are true, assertions may be made with substantial strength. But in almost all real systems, many of these things are not true. As a result, the arguments are far weaker than they are often assumed to be. While the examiner may use such information to attribute actions to actors or processes to mechanisms, the assumptions and events that allow these traces to be asserted as causal should be stated, and to the extent that they are not stated, the attributions are subject to challenge. If challenged without adequate understanding by the attributing examiner, they may be problematic.

Sensors used for other purposes and related approaches

Just as the methodologies used in intrusion and anomaly detection and detection of undesired content are problematic for generating defined measures of similarity, they are problematic in attribution of effects to causes. But just as they may lead to investigative approaches that may ultimately be successful in identifying relevant information in other areas, they may be helpful in attribution. For example, a set of detectors within an environment may provide sets of traces that are redundant, not normally alterable by the same parties that have access to the normal causes of those traces, and use different methods and sources to generate their traces (e.g., network traffic vs. activities on computers vs. log files with records

from programs vs. records from authentication servers, etc.). These redundant traces may provide additional information that fills in missing steps in the trail from cause to effect and may eliminate or reasonably cover many of the possible paths from cause to effect that otherwise might exist.

A good example of this might be the attribution of actions to actors. Returning to the previous example of showing that a particular individual was continuously or nearly continuously present at a particular location during a particular time period, and therefore performed certain activities, (from Chapter 6) the traces from independent systems and mechanisms may be used to eliminate one after another of the alternative possibilities. They may also allow the examiner to confirm the hypotheses that allow attributions with reasonable certainty. Here are the hypotheses from that example, taken in light of the use of independent sensor traces.

- *(1) There may be events that demonstrate that the individual of interest was the only individual present in the relevant facility over the time frame*

  Independent traces may indicate that nobody else entered the room from which the activities took place, that the only detected traffic engaging another system communicating with "source" system had consistent properties to "source", that the programs run on the computer in question were authenticated using the authenticators of the identified individual, and that authentications were performed at regular intervals using the authentication device that was assigned to the individual. The traces may also show that other individuals who were in nearby areas were all using their systems over the same period, that nobody entered rooms or left their work areas during that period, that the individual with the identified authentication device paid for a particular meal at lunch using that device, and that other workers saw them eat that meal.

- *(2) Traces may give no indication of any system subversions and be consistent with proper and normal operation over the period of interest;*

The independent traces may show that no traffic, other than traffic associated with the identified applications and to the identified locations for those applications, took place over the relevant periods, that the patterns of keystroke timing indicated ongoing use with behavioral characteristics consistent with those historically recorded for the individual, that the sequence of uses was consistent with the usage patterns of the individual, that external verifications of the computer and its operations were consistent with normal operations, and that periodic scans and other similar methods used to confirm proper configuration of the system were consistent with the system operating normally.

- *(3) events may stipulate that the traces are legitimate, taken from the specific system at the times indicated, and that the system operated with a standard version of a particular operating environment;*

  *S*tatements by the owner's representative who was tasked with configuration and management of the systems might authenticate these as facts and provide records of specific dates and times with signatures indicating that these activities were planned, performed, tested, and verified.

- *(4) The quantity of content and usage patterns reflected in traces of various activities such as the number of characters entered into a document over a defined time frame and Internet access traces such as the Web browser cache, may show that every few seconds some activity was performed.*

  For example, the independent traces may show that Web access from the identified machine is consistent with the information contained in the Web cache, that emails sent to other parties during that time included attachments that contain document files that can be independently sought from email servers and recipients, that ongoing exchanges were underway with a particular service such as a database interface that takes input and produces output based on screens that the user normally uses in an interactive mode, that those exchanges showed activities at defined times, and that the results of these activities was used in subsequent

exchanges with other services, or demonstrated within a document.

Each of these sources of traces provide independent confirmations of a hypothesis about the attribution of actions to actors, and as such, they are good indicators of causality. But they can be challenged. For example:

- These mechanisms may all operate by the observation of network traffic, and that traffic may be readily forged, replayed, or otherwise generated or subverted.
- The mechanisms themselves might be operated by another individual who forged those records and had a history of disputes with one of the parties.
- There may be anomalies in the traces associated with those mechanisms that causes them to be less credible as sources.
- A lack of separation of duties in the operation and administration of systems may be an issue in this case.
- Mechanisms may not record all traffic, but only select traffic.
- This traffic could reflect a third party carrying out these activities in the background from a different location while the user was present, or not, for different periods.
- Intrusion detection and other similar systems are subject to subversion and false information.[248,249,250,251]
- What we see in observing the outputs of these systems is not always a true reflection of what actually took place.

248 F. Cohen, "National Info-Sec Technical Baseline: Intrusion Detection and Response", Lawrence Livermore National Laboratory and Sandia National Laboratories, December, 1996. Also available at: http://all.net/journal/ntb/ids.html

249 F. Cohen, "A Framework for Deception", F. Cohen, et. al. 2001, available at: http://all.net/journal/deception/Framework/Framework.html

250 F. Cohen, I. Marin, J. Sappington, C. Stewart, and E. Thomas, "Red Teaming Experiments with Deception Technologies", 2001, available at: http://all.net/journal/deception/RedTeamingExperiments.pdf

251 F. Cohen and D. Koike, "Leading attackers through attack graphs with deceptions", IFIP TC.11 Computers & Security, Volume 22, Issue 5, July 2003, Pages 402-411, at http://all.net/journal/deception/Agraph/Agraph.html

One of the advantages of the various sensor mechanisms that exist within many environments is that they are usually admissible as evidence because they produce normal business records, relied upon for their accuracy in day-to-day operations, and they tend to be independent of the systems they observe.

Fusing redundant sources of data

The examples from sensor data approaches can be extended to the more general approach of using redundant sources of data to increase the certainty of interpretations that associate traces and events with each other and allow causal chains to be established. Multiple sources of traces that are independent, separate, different, and at different places in the chain of events that constitutes the demonstration of causality may be very helpful in attribution. The ability to correlate them in time or other ordering processes and relate them to each other establishes an increasingly useful causal chain that may be used for attribution, if the end points or other content are adequately strong, in light of circumstances, to establish the attribution to the level of proof required for the standard of proof in the case at hand.

The typical process of fusing events and traces involves creating an ordered sequence of traces and events that are relevant to each of the causal chains of interest. If the traces and events allow an ordering that has distinguishable causes prior to effects, if there are adequate linkages between these traces and events to allow demonstration consistent with being from the same causal chains, and if other traces or events that should be present to show other causal chains of the same sort, are not missing, then this can be used to establish the causal chain and attribute actions to actors to the extent that the causal chain reaches between different things of interest to the case. The chain may not reach all the way to a human being or other physical events, but it may reach far enough to be of use.

The complexity of creating causal chains is not known at this time, and it may be that there is no general solution to its complexity. The fundamental problem is that there is no way to identify what all of the possible linkages may be between causes and effects. While information physics tells us that we can drive any system forward

given a totality of its history, it also tells us that without this total history, it may be impossible to uniquely derive an entire causal chain, even in the forward direction. In the reverse direction, the increasing number of possibilities drives complexity up in the size of the set of all possible sequences producing known traces.

This approach is also problematic in that, in order to make these sorts of assertions, it would be necessary to show the reliability of the methods being applied. Previous experiments on time sequencing in computers have shown that the normal sequences are not perfect, at least as to placement of portions of files on disks and the different meta-data associated with files in file systems.[252] The examiner wishing to make this claim should spend some time doing appropriate experiments to show the validity of the technique before trying to use it is a case. Chapter 8, discusses this further.

How content comes to be as it is

In most general purpose computer systems, computer programs operate as sets of concurrently executing FSMs, and often interact with each other directly, through network interfaces, through process mechanisms, and through file systems. The many possible sequences of events, FSMs, and sources of input, can make the situation extremely complex. But in most normal operating conditions, a relatively small number of programs are used, they take inputs from a limited set of places, interact with a limited number of other programs, files, and systems, and do so in relatively structured ways.

As a result, much of the content produced by most modern computers involves taking input from sources and transferring it to destinations while adding or altering indicators of the source, destination, travel path, and/or handling. There are various common transforms used by different programs, such as changing character sets, line formats, adding or removing various delimiters, putting content into and taking them out of datagrams, and so forth. Some programs do other sorts of processing, like searching for related data and combining the original data with the related data,

252 S. Willassen, "Methods for Enhancement of Timestamp Evidence in Digital Investigations", Doctoral thesis for the degree philosophiae doctor, Trondheim, January 2008, Norwegian University of Science and Technology.

performing calculations on data and providing results of those calculations, looking up entries in databases and providing related material based on the inputs, and so forth.

In seeking to understand how traces come to be as they are, many examiners make assumptions about the normal operation of systems and use those assumptions to associate different content within traces with different sources and methods by which they might normally appear. For example, they may identify different parts of traces with inputs, or seek similarities between traces based on similar apparent sourcing, travel patterns, receptions, destinations, or handling. While such assumptions are commonly made, this sort of attribution has substantial underlying assumptions that are problematic in cases where parties seek to subvert systems. This includes both subversion by the parties to the legal matter and subversion by independent parties for their own reasons and using their own methods.

Provenance and attribution in the digital world

Provenance, as typically defined, stems from the Latin word "prōvenīre", which means "to come forth", (pro-, convene, -ant). More recently, it is associated with identification of the origins and path by which something came to be wherever and whatever it is.

Chain of custody

In legal settings, there is the notion of "chain of custody", the sequence of holders of a particular piece of proffered material that is to be introduced as evidence. The chain of custody typically starts at the point in time when the act of interest (crime, violation, or other relevant act) took place, and carries through to the identification of a record or other item related to the act as possible evidence, its collection, preservation, storage, and transportation, and ultimately, its introduction as evidence in court.

The chain of custody is typically documented at steps along the way as to where the item is, who handled it in what way, where and how it was stored, and when each act in the chain of custody took place. The people who handle the item are typically available to testify as to exactly what they did, how, and so forth, and are subject to cross examination.

The perspective of this book assumes that traces provided to the examiner are a "bag of bits" (i.e., digital traces), and not physical evidence. From a process standpoint, the examiner may be given original evidence at some point. While outside of the scope of this book, clearly, when such an event occurs, the examiner must retain such chain of custody required for the matter at hand, typically making a forensically sound copy of the traces and returning the original evidence to a proper storage facility.

Examinations and provenance

In performing examination, the process by which an item was examined and turned into another item (e.g., a printout) is also subject to the requirements of provenance, and the results potentially subject to chain of custody, to the extent that they cannot be independently verified by the opposing side from the original writing from which they were derived. This goes to the issue of tools, process, keeping contemporaneous notes, and related matters. As a practical matter, this is covered rather more deeply in other works,[253] but some attention is warranted here.

In performing examinations, it is generally appropriate to take notes of activities performed, tools used, and results produced, and to provide these details, when so ordered by the courts. In US Federal proceedings, for example, expert reports are supposed to contain specific things, to wit, in pertinent parts:[254]

> "(F) Expert Witness Reports:
>
> (1) ...any party who calls an expert witness shall cause that witness to prepare a written report for submission to the Court and to the opposing party. The report shall set forth the qualifications of the expert witness and shall state the witness's opinion and the facts or data on which that opinion is based. The report shall set forth in detail the reasons for the conclusion,... Additional direct testimony with respect to the report may be allowed to clarify or emphasize matters in the report, to cover matters arising after the preparation of the report, or otherwise at the discretion of the Court. ... An

253 F. Cohen, "Challenges to Digital Forensic Evidence", ASP Press, 2008.

254 US CODE: Title 26 App. Title XIV, Rule 143 "Evidence", subsection (f) "Expert Witness Reports".

> expert witness's testimony will be excluded altogether for failure to comply with the provisions of this paragraph, unless the failure is shown to be due to good cause and unless the failure does not unduly prejudice the opposing party, such as by significantly impairing the opposing party's ability to cross-examine the expert witness or by denying the opposing party the reasonable opportunity to obtain evidence in rebuttal to the expert witness's testimony."

The facts and data that must be included in the report, includes all of the relevant results of examination about which conclusions may be drawn and expert testimony given. This implies that the details of how, when, who, and what associated with all potential evidence must be retained to establish the provenance of results, including without limit, any underlying scientific methodologies, and any information needed to show that the methodologies were properly applied in carrying out examinations and producing opinions.

Since all of this is required in any such report, it, in some sense, makes good sense for the examiner to start writing the report as they carry out their examination activities and take any notes related to their activities directly within the report. For example, a report might start by indicating that:

> On or about [Date] at [time] I received the following items:
>
> - A file named "..." and indicated to contain ...

This then provides provenance information as a contemporaneous record that is part of the report itself, and assures both that the report is accurate and that it contains the necessary information.

Provenance as part of attribution

In terms of attribution, the examiner has potential responsibility for establishing and properly verifying provenance issues with respect to the traces and events taken into consideration in the attribution process. Typically, this is done technically by using the consistent properties and features of the traces, and in conjunction with events, by tying traces and events together. For example:

- An event involving an admission in court by a party
- An event involving documents resulting from a subpoena

- Traces involving information relating to those events

In attributing actions to the actor, the examiner in such a situation may report, as an example, that the admission by the party shows that they were responsible for a computer attached to a particular IP address at a particular time, while the documents might reveal that that IP address was used in that time frame to create an account, and the traces may be used to demonstrate that the content of that account was identical to the content released in a subsequent activity by the party.

The DFE in this case is only the match between the released content and the content of that account, but the attribution combines the events with the traces to draw a larger conclusion, that "it appears that [party] was responsible for the release of that information". Of course this is a simplified version that leaves many questions, such as how the time frames were established, why that is the only possible source of that content, and how that shows that the party actually undertook those actions as opposed to some other party.

This then gets to the issue that provenance information is, in most cases, far from perfect. There are almost always potential ways in which it may be challenged, and it is to be expected that the opposition will challenge it, as is their duty. And this is, in large part, why particular language is appropriate in making provenance-related conclusions. The phrase "it appears that" was chosen for a reason.

Attributing actions to human actors

There have been many announcements, papers, discussions, and claims regarding attribution solutions, but few literature reviews and little useful technology. Cutting to the chase, content inspection for attribution does not work well, if at all, today. Under naïve deception, these methods perform no better than random guessing. Without deception, for 5000 words of prose per individual and collections of 10 to 20 individuals, current methods correctly identify authorship, at best 80% of the time, and at worst, never. Current approaches are summarized here considering potential use in legal proceedings, terminations, authentication, and attack attribution.

Using authentication for attribution

Authentication is intended for use in confirming an identification. As such, its intended purpose is not attributing actions to actors. But it is often used as a basis for attribution because it provides some level of demonstration that an individual is present at a system at a time. The question for the forensic examiner seeking to use this to attribute actions to actors is what that level of certainty is, how it is determined, and whether it is adequate to the case requirements.

Authentication methods typically have very low probability of error in statistical terms. To the extent that they involve authentication devices, like hardware tokens, or memorization, like passwords, they have an extremely low probability of “false” authentication. For example, even a 4 character password has a 1 in 26^4 (just over 2 in a million, or .0002%) - chance of being guessed randomly. For false negatives, authentications with these methods essentially never fail except through typographic errors by users, which happen more often for longer and harder to guess mechanisms. However, authentication is not attribution.

A relevant question for the forensic examiner is how and to what extent authentication can be used as a reliable tool for attribution based on traces and events. There are several hurtles to pass before considering the details of the methods, the first of which is to determine whether the traces are available to try such an interpretation. Many of the methods described in this field involve things like collecting data with precision and accuracy at the level of milliseconds, or require that previously implemented and calibrated devices be in place and properly functioning. This is typically unavailable in internal audit logs and other similar traces found within computers.

Unless the authentication method was undertaken as part of normal business practices, it is unlikely that the necessary information will be available to the forensic examiner. But some traces may be found in detailed network logs, from incidental packet sniffing, or from company keystroke loggers used to track user behavior. Some less reliable methods of attribution may also be available. But the question remains: "How can they be reasonably applied by the forensic examiner?"

Types of authentication methods

Authentication has historically been categorized as consisting of something the user has (an authentication device), something the user is (a biometric or behavioral property), or something the user knows (a password, pass phrase, or query response). Because biometrics and behavioral methods are really attribution methods used for authentication, they will be handled separately in this review.

Something the user has

There are many other approaches based on something the user has that are used for authentication, and these may also be used to try to attribute actions to actors. But at best, on their own, they can only attribute actions to actors possessing the authentication mechanism, and not to the individual actors themselves. It has been suggested that the only way to show that people are present is to have a continuous video with time details of them siting there, and the only way to make certain it was their use is to have a camera watch all of their actual typing and other input device uses while also recording the display outputs. This of course depends on the standard of proof in the matter at hand and other events and traces that are available.

For something the user has, the examiner should be careful to point out that use only indicates that the device was used or forged, and not that the individual, or even the device, was present. The link to individual presence must be made by some other means.

Something the user knows or can do

Passwords, and variations on them, have been in use since biblical times. While modern variations may include behavioral indicators and query response systems, these are really no different than they have been for millennia. Many flaws and limitations with these systems are widely known and have been widely published over many years. Variations on approaches that use human ability to do things have also been introduced.[255]

255 F. Cohen, "Algorithmic Authentication of Identification"', Information Age, V7#1 (Jan. 1985), pp 35-41.

Some major problems with these sorts of authenticators from a forensic standpoint include, without limit;

(1) the actual sequence used to authenticate is not usually available to the DFE examiner, since it is typically not stored;

(2) the records of a successful authentication may be easily forged;

(3) authenticators are often easily guessed or replayed and regularly used by malicious third parties to forge authentications;

(4) many automated mechanisms use these sorts of authenticators and can readily perform tasks in place of the individual;

(5) users commonly share identities and authenticators with each other or use group accounts;

(6) passwords are easily observed by others, both directly over the shoulder, and in recording devices such as keystroke logging devices, videos of user activities, and so forth;

(7) the large number of passwords to different systems in use today result in stored copies and widespread reuse; and

(7) just because an individual authenticated does not mean that they were the individual who did the subsequent acts or even that they remained present.

Biometrics and their failure rates

Biometrics (e.g., fingerprints, retina or iris patterns, hand geometry, facial appearance, heat signatures, foot falls, etc.) are often used in computers as a method to authenticate users. As such, they fit nicely into the human attribution arena. While they are used for authentication, they provide authentication precisely because of their qualities associated with attribution, and they are useful for authentication, only if and to the extent that they provide few false positives and negatives. Biometric authentication methods have shown false acceptance rates as follows:[256]

256 T. Ruggles, “Comparison of Biometric Techniques” Revised 2001-07-10, http://www.bioconsulting.com/bio.htm

Retina Recognition	1 to 10,000,000+
Iris Recognition	1 to 131,000
Fingerprint Recognition	1 to 500
Hand Geometry Recognition	1 to 500 (from small group)
Signature Dynamics	1 to 50
Voice Dynamics	1 to 50
Facial Recognition	No Data Available
Vascular Patterns	No Data Available

In addition, each of these approaches has other error rates. False acceptance only rates the likelihood of accepting an authentication based on the biometric from a small population. There are also other sources of error, like data collection errors, in which the original information subsequently used for authentication is not done correctly or is of a different individual. The Automated Fingerprint identification System (AFIS) system had more than a 98% correct identification rate and a false positive identification rate of less than 1% across the entire population it covered (over 100 million individuals). Signature and voice dynamics have input problems, in that measurements made (e.g., stroke direction, pressure, acceleration, length, and vocal frequency distribution, rate of frequency change, etc.) are not consistently measured to high precision and accuracy by the mechanisms that measure them. Database matching is also imperfect for these methods. Independent statistical data newer than 1997[257] was not found in the literature, despite many publications and claims regarding these methods and technologies.

It is also important in understanding the use of these methods in digital systems to take into account that they are normally used for authentication of identified individuals from known populations of limited size under controlled conditions where the mechanism is previously calibrated for the purpose. This is fundamentally different from their use to identify an individual or attribute physical presence at a place to an individual, which is typically the desired use in non-digital forensic examination and investigations.

257 J. Holmes, L. Wright, R. Maxwell, “A Performance Evaluation of Biometric Identification Devices”, (Sandia National Laboratories, SAND91-0278/UC-906, June 1997). is cited in many sources as the real basis for this data.

Their suitability for identification is dubious in the general sense. But if there is a small population of possible suspects with known characteristics, and if the traces are available to do the analysis with the appropriate level of calibration, their results may be leveraged after the fact to confirm that the traces are consistent with use by known subjects, or that they are more consistent with one subject than other subjects. Thus they may be used for individualization in some cases through the process of elimination.

The quality of conclusions regarding identification and attribution should be subject to scrutiny and validation. The actual data collected and available in traces typically does not include the raw data collected. This is typically obfuscated during use and destroyed after use to prevent its exploitation for false entry. The the only digital traces typically available are the result of an authentication process indicating that a biometric was checked and passed or failed for this particular purpose. In addition, the presence of an individual at a system does not imply that they were responsible for the actions of that system.

Behavioral methods

Behavioral methods (e.g., keystroke analysis, word usage patterns, typing errors, spelling errors, the commands used in the order applied, how editing is done, etc.) are used, often experimentally today, to authenticate known users with previously collected characteristics under controlled circumstances. In the larger picture of attribution, human behavior forms an increasingly explored area. Observables like keystroke sequences and patterns, word usage patterns, typing errors and quirks, spelling errors, the commands used in the order applied, how editing is done, and other similar things have been increasingly explored as attribution indicators.

Fist, Keystroke patterns, Footfall, and related approaches

For a long time, it has been conjectured that analysis of keystroke timings, writing patterns, and other similar phenomena might be used to attribute actions to actors. Starting at least in 1980,[258] extensions of the work in World War 2 for identification of the "fist"

258 Gaines, R.S., et al. 1980. Authentication by keystroke timing: Some preliminary results. Rand. Report R-256-NSF. Rand Corporation. Available at http://www.rand.org/pubs/reports/R2526/.

of key code operators was considered for identifying which of a group of known individuals were typing on a keyboard attached to a computer. While the early work in this arena was promising, it took quit a while for progress to be made, and that progress revealed many limitations of such processes.

Authentication based on behaviors, such as keystroke patterns, are easily forged with methods similar to those used to collect the keystrokes in the first place.[259,260] But even if they weren't, tests that indicated 93% correct recognition on average (i.e., only about 1 in 14 legitimate uses would be declared illegitimate), were only 51% correct for some users (half the time they would be declared illegitimate), and these results were only under very limited conditions (i.e., for distinguishing individuals from groups of about 50 people, with entry of 200 or more characters, use of the same keyboard by each user each time, using entry of free text of their own composition, with accurate timing information to the millisecond, including key press duration, transition times, and special key uses, and collected in structured tasks using identical software and operating environments installed on every computer).[261]

Some early IBM-Selectric typewriter work[262] and the secure shell timing attack work have demonstrated some exploitation of typiing behaviors.[263] For keystrokes with millisecond timing resolution, password only timing, and long training times, failure rates (false

259 F. Cohen, et. al. "Leading Attackers Through Attack Graphs with Deceptions", IFIP-TC11, `Computers and Security', V22#5, July 2003, pp. 402-411(10).

260 F. Cohen, I. Marin, J. Sappington, C. Stewart, and E. Thomas, "Red Teaming Experiments with Deception Technologies", 2001.

261Mary Villani, Charles Tappert, Giang Ngo, Justin Simone, Huguens St. Fort, Sung-Hyuk Cha, "Keystroke Biometric Recognition Studies on Long-Text Input under Ideal and Application-Oriented Conditions", School of CSIS, Pace University, Pleasantville, New York, 10570, 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06).

262 Gaines, R.S., et al. 1980. Authentication by keystroke timing: Some preliminary results. Rand. Report R-256-NSF. Rand Corporation. Available at http://www.rand.org/pubs/reports/R2526/.

263 Song, D., et al. 2001. Timing analysis of keystrokes and timing attacks on SSH. Proceedings of the 10th USENIX Security Symposium. Available online at http://www.usenix.org/publications/library/proceedings/sec01/song.html.

acceptance) of 1 in 20 (95% recognition) were achieved. False rejections are also an issue and as false acceptance decreases, false rejection increases.[264] This seems unlikely to satisfy the requirements of forensic examination,[265] and would be extremely problematic for larger groups under less controlled circumstances.

Movement patterns from mobile devices, user gait patterns, Web click patterns, and writing patterns from pen computing have also been tried,[266] and cognitive methods suggested.[267] But we have yet to see successful attribution with these methods in the peer reviewed literature.

Suitability for identification is dubious, but for a small population of suspects with known characteristics, if the traces are available with appropriate calibration, results may be leveraged after the fact to confirm that traces are consistent with known subjects, or more consistent with one subject than other subjects. Again, such traces are rarely available after the fact, even when they are collected for authentication.

Stylometrics, phrasing, and similar document analysis

The basic premise of stylometrics and phrasing analysis is that people are formed over time by how they acquire knowledge, skills, and techniques from the people, places, societies, and media they interact with. These trained and learned skills and behaviors take time to develop, and individuals change behaviors as they develop over time. The fact that actors can only use the methods they are aware of, the notion that methods are commonly originated and disseminated through a path, and other similar notions, provide the means to potentially reverse the paths, identify properties, type and

264 Aykut Guven and Ibrahim Sogukpinar, "Understanding users' keystroke patterns for computer access security", Computers & Security, Volume 22, Issue 8, December 2003, Pages 695-706.

265 B. Rao, "Continuous Keystroke Biometric System", Media Arts and Technology, September 2005, A University of California, Santa Barbara for partial requirements of Masters of Science in Media Arts and Technology.

266Padmanabhan, B. and Y. Yang. Unpublished manuscript. Clickprints on the Web: Are there signatures in Web browsing data? Available at http://knowledge.wharton.upenn.edu/papers/1323.pdf?CFID=720523&CFTOKEN=57530247.

267F. Cohen, et.a,. "A Framework for Deception", http://all.net/journal/deception/Framework/Framework.html

particularize sources by characteristics, and perhaps individualize a source. This approach falls under and is compatible with the notion of identifying attacker type[268] and individualizing based on specific characteristics such as capabilities and intent. A reference base is built and pattern recognition is done on traces against baselines.

Stylometric[269] methods used for disambiguation of word sense and plagiarism detection have been admitted in US courts in select cases, but they are readily susceptible to deception.

N-Gram Analysis and Other Statistical Methods

In this approach, collections of one or more n-tuples of symbols in sequence are used to characterize traces. Unigrams (1-grams), bigrams (2-grams), and trigrams (3-grams) are most commonly used, but this can be expanded to n-grams in the general case. The process typically starts by splitting traces into tokens, which is essentially a syntactic analysis as described in Chapter 5. This low-level parsing into tokens can be characters, bytes, bit sequences, or other parsed entities, depending on how the behavior is reflected in the traces. Statistics, like the count of each symbol in the symbol set based on Shannon's information content[270] or similar measures as described in Chapter 5 may be used.

2-grams are pairs of symbols, and a number of methods have been applied to trying to correlate pairs to sources. This includes, without limit, dice coefficients, Fisher's exact test (both left sided and right sided), log-likelihood ratio, mutual information, point-wise mutual information, odds ratio, phi coefficient, T-score, Pearson's chi squared test, and any number of other methods from standard statistical analyses. 3-grams and higher order sequences have also been examined with similar statistical methods, including without

268F. Cohen, C. Phillips, L. Swiler, T. Gaylor, P. Leary, F. Rupley, R. Isler, and E. Dart, "A Preliminary Classification Scheme for Information System Threats, Attacks, and Defenses; A Cause and Effect Model; and Some Analysis Based on That Model", Sandia National Laboratories, September, 1998, also in Computers and Security and The Encyclopedia of Computer Science and Technology, and at: http://all.net/journal/ntb/cause-and-effect.html

269 C. Chaski, "Who's At The Keyboard? Authorship Attribution in Digital Evidence Investigations", International Journal of Digital Evidence, V4#1, 2005.

270 C. Shannon, A Mathematical Theory of Communications, Bell Systems Technical Journal. 3, no. 27, (July 1948).

limit, N-gram largest token size of value, hidden Markov models, morphemes and phonemes, gender identification, authorship attribution, bag-of-word techniques, and non bag-of-word similarity techniques. Additional data and references are available to the interested reader.[271] Similarity analysis may help create larger sets of N-grams by identifying greatest common factor groups of examiner-identified features and characteristics.[272]

Attack attribution

When looking at characteristics of malicious computer attackers, the work has focused on attack methodologies, but similar approaches may be taken to normal user behavior. These are accomplished by; log file analysis, which is often directly available from traces; attack graph comparison, which is far more complex to understand from traces, but has been done through the addition of sensors and analysis of exiting and additional sensor data; identification of attack code similarities either directly by code examination or by examination of traces produced by attack codes; and by unique content used in attacks, such as keys for cryptographic systems, arrangements of memory within programs, and so forth.

Additional insight has been gained by considering the sophistication and usage patterns of attackers, and the same basic approach has been used to try to identify individuals in text forums, such as Internet relay chat (IRC) postings or postings to newsgroups.

Attribution has been studied for various purposes, including its use in trying to attribute attacks to the attackers that carry them out. In one review of the issues[273] an approach to identification of the individual responsible for an attack or other activity, known as level 3 attribution, is differentiation by how symbols are used to communicate. This leads to several general methods.[274]

271 Survey / Analysis of Levels I, II, and III Attack Attribution Techniques", available at: http://www.cs3-inc.com/arda-survey.pdf

272 F. Cohen, "Identifying and Attributing Similar Traces with Greatest Common Factor Analysis", Pending publication.

273 C. Uber, Personal correspondence, Dec. 2003 - Feb. 2004.

274 Survey / Analysis of Levels I, II, and III Attack Attribution Techniques", available at: http://www.cs3-inc.com/arda-survey.pdf

Limitations of human attribution

Human attribution is problematic in normal circumstances, and under deception, the problems increase.

Limitations of human attribution in normal operation

The basic challenges to attribution in normal operation are that:

- The underlying theories and methodologies are not widely accepted or well vetted in the scientific community.
- Many people behave similarly to within close tolerances,
- There are about 6 billion people alive today, most of whom are candidates for attribution in the Internet age,
- Behavior isn't so consistent and unique that it is easily differentiable based on ready observables,
- Sensor and analysis precision, accuracy, stability, and capabilities limit the ability to measure and analyze
- Analysis capabilities are normally based on methodologies not intended for forensic purposes or high surety levels,
- The mechanisms used to collect, process, transport, store, and analyze are not always reliable or adequately protected, and
- Even if the methods identify an individual who was in fact present at an identified place at an identified time, this is not adequate to establish that the behaviors of the system were caused by that individual.

Analysis addressing attribution of attacks[275,276] characterizes the alternative approaches to level 3 attribution.[277] Table 7.1 summarizes one assessment of the state of the art circa 2004.

275 D. E. Denning, "Cyber Attack Attribution: Issues and Challenges", March, 2005 presentation from "Center for Terrorism and Irregular Warfare - Department of Defense Analysis - Naval Postgraduate School".

276 K. Narayanaswamy, "Survey/Analysis of Levels I, II, and III Attack Attribution Techniques", Cs3, Inc., April 27, 2004.

277 Survey / Analysis of Levels I, II, and III Attack Attribution Techniques", available at: http://www.cs3-inc.com/arda-survey.pdf

Measurable & Method	*Characteristics Uncovered*	*Effectiveness*
Document analysis - natural language methods	Attacker goals, style, education, native language, knowledge, comparison to prior writings	Computationally intractable, but potentially more accurate
Document analysis - statistical methods	Attacker goal, style, education, native language, knowledge, comparison to prior writings	Tractable; probabilistic answer provided; attacker might be able to deceive analysis
Keystroke timing	Comparison to prior profiles; left-handed or right-handed	Tractable, but results unreliable – attacker can mislead
Email authorship	Similar to natural language; gender	Potentially useful – similar problems to document analysis
Attack code analysis	Attacker's sophistication level; tools used; knowledge; capabilities and resources	Potentially effective; no need for cooperation with anyone else
Attack models	Enumerate potential paths for attacker to take to perpetuate activity	Starting point for level 3 attribution process

Table 7.1 - Level 3 attribution methods, characteristics, and effectiveness

The underlying theories and methodologies discussed above, particularly those related to behaviors, are not widely accepted or well vetted in the scientific community. In many cases there have been one or two studies, the source code and analysis methods have not been published to the level of detail required to confirm or refute the claims, standard sample data sets are not available for testing, and many potentially relevant factors have not been studied. Even with these limitations, most of the techniques have error rates that leave many candidates in many situations, and produce many false positives and negatives.

A population-related challenge is also present because, for most Internet-connected computers, billions of people now have potential access. With this many candidates, there are many individuals who may have very similar behavioral characteristics, and inadequate studies have been done to characterize this. The tolerances required for differentiation in large populations are very tight, and in most cases, far tighter than the current tolerances of the technologies at issue.

Behavior also isn't so consistent and unique that it is easy to differentiate between individuals. Inadequate studies have been done to demonstrate such a capability at all, and the standards for fields like psychology and other behavioral sciences are typically limited to 95% certainty that correlations exceed those required for the null hypothesis. Even these standards are higher than most of the results above, and psychology results to date don;t support an adequate basis to believe that individualization is feasible based on the characteristics studied.

Sensor and analysis precision, accuracy, stability, and capabilities limit the ability to measure and analyze. Current sensors are largely in place for purposes other than attribution, and are not designed with these sorts of characteristics in mind. Thus the readily available data is typically of poorer quality that the best data available from experimental results shown above.

Analysis capabilities are normally based on methodologies not intended for forensic purposes or high surety levels. For example, many are based on support vector machines or similar clustering analysis techniques that are intended to work for tasks where humans can override, must confirm, and so forth. They are not intended for or designed to support forensic analysis.

The mechanisms used to collect, process, transport, store, and analyze are not always reliable or adequately protected. In fact, many of the techniques identified above are designed to support investigation of misuse of or attacks on computer systems and networks. The reason this data is being gathered and analyzed is that systems and networks are highly vulnerable and constantly subject to malicious attacks that subvert their normal operation. In other words, the information systems that provide the traces used

in analysis and attribution may be unreliable and the mechanisms they contain are commonly circumvented. Any attribution approach in this realm will necessarily have the potential for being unreliable and inaccurate because the traces are potentially unreliable because the system measuring or generating the traces may not be operating properly. Most current mechanisms used for storing, analyzing, and making decisions based on these methods are also susceptible to attack and usually readily defeated. Even if an individual in question was present, the system may not have actually been under their control, and the records of their presence may not accurately reflect what actually took place. If there was an attack on the system the user was using or the system or records, the attribution information may be irrelevant.

In order to demonstrate attribution of actions to human actors at computer systems, it could reasonably be argued that the systems have to be demonstrated to properly reflect the actions of the actor in their record-keeping. While normal business records may apply to allow evidence to be admitted, attribution without demonstration of reliability is certain to be problematic. Since there is normally no way to show that a system was not under attack, and because of the large numbers of reported attacks on systems that produce the capacity for remote attackers to alter operations, records, interfere with normal use, and appear to be the individual at the keyboard, this area is particularly opaque. Add in the malicious insider altering records, and the problem of human attribution becomes far harder.

Limitations of human attribution under deception

If there is a possibility of intentional forgery, these issues get more complex. It is relatively easy and commonplace for novices to carry out simple forgeries, like making it look like an electronic mail came from a source it did not come from. Moderately skilled malicious actors can easily carry out attacks that bypass authentication methods, forge log files in systems that don't maintain them securely, create false identities within computer systems and authenticate to those identities, steal user identities and passwords and reuse them, or any number of similar things.[278]

278 F. Cohen, C. Phillips, L. Swiler, T. Gaylor, P. Leary, F. Rupley, R. Isler, and E. Dart, "A Preliminary Classification Scheme for Information System Threats, Attacks, and Defenses; A Cause and Effect Model; and Some Analysis Based

Sophisticated attackers with financing, access to research and development, and other similar assets, can do far more complex and realistic forgeries such that many redundant indicators will fail to differentiate a forgery from an original. For example, over a network, keystroke activity with timing data and other related behaviors have been emulated in deception-related experiments to good effect.[279]

In addition, insiders (authorized individuals not faithfully and loyally carrying out their duties) are found in case after case to have altered computer records, used other individuals' accounts, and forged ownership as part of their methods.[280,281] Such actors commonly alter records to attribute actions to other actors, often including other workers against whom they hold a grudge.

Recent results suggest that naïve deception readily and effectively defeats many stylometric approaches.[282] This study looked at simple deceptions and ran comparative analysis with 27 attribution methods. The sample set had 15 authors with 5,000 words from each, 500 involving imitation and 500 involving obfuscation. A total of 1080 experiments were performed applying different methods to different sample sets, each examining all 12 obfuscated essays and 12 imitative essays against all possible training documents, with a resulting chance-level performance of 1 in 12 correct.

Under obfuscation, where individuals sought to change their writing styles, the best performance was correct in its classification 42% of the time (5 in 12), and the worst was never correct. For imitation,

---

on That Model", Sandia National Laboratories, September, 1998, also in Computers and Security and The Encyclopedia of Computer Science and Technology, and at: http://all.net/journal/ntb/cause-and-effect.html

279 F. Cohen, I. Marin, J. Sappington, C. Stewart, and E. Thomas, "Red Teaming Experiments with Deception Technologies", 2001, available at: http://all.net/journal/deception/RedTeamingExperiments.pdf

280 M Keeney, E. Kowalski, D. Cappelli, A. Moore, T, Shimeall, S. Rodgers, “Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors”, Jan 2005.

281 E. Shaw, K. Ruby, and J. Post, “The Insider Threat to Information Systems - The Psychology of the Dangerous Insider ”, Security Awareness Bulletin, No. 2-98, 1998.

282 P. Juola and D. Vescovi, “Analyzing Stylometric Approaches to Author Obfuscation”, Advances in Digital Forensics, IFIP TC11.9 Digital Forensics Conference, Orlando, FL 2011-01-31 - 2011-02-02. pp115-125 (Chapter 9)

where samples of other writing styles were provided and imitation was sought to deceive authorship, the best performance was 25% (3 in 12) correct attribution, and the worst was never correct. No method performed significantly better than chance at the 95% certainty level normally used for the social sciences.

Forgery of biometrics has also been widely demonstrated. Eye recognition forgery is somewhat problematic, but fingerprint mechanisms have been readily overcome with gelatin imprinted with ridges and placed over human digits, hand geometry can be readily forged with a mold, signature forgery with timing and stroke information can be automated using computer output devices, voice recognition is susceptible to various recording and playback mechanisms, and facial recognition is problematic when people smile and can be overcome with simple masks.

The problem with almost all of these approaches is that, while they are good ideas, and some of them have been significantly explored over the years,[283] from a DFE attribution standpoint, there is little definitive information that can be used today to associate reliability with these methods. Significant study is needed to get a good set of relevant metrics for reliably allowing these approaches to be used for more than limited confirmation of an individual from a small group without forgery and with admitted attributed baseline samples.[284] Under intentional forgery they fall apart to the point where they approach random guessing.

Indicators as opposed to attribution

Clearly, there are many approaches to attribution in the case of malicious attacks, and many of these approaches may be applicable to available traces and events in legal matters not related to attacks. But the certainty with which these approaches

283 M. Corney, "Analysing E-mail Text Authorship for Forensic Purposes", Masters Thesis, Queensland University of Technology, March, 2003 [This thesis examines using a variety of classifiers with output fed into a Support Vector Machine (SVM). The approach is to compare a specific email to an SVM model built from a corpus of emails with known provenance e.g. given 20 emails from each of A, B and C, compare a new email to those models to see which author it is most likely to belong to.]

284 C. Chaski, "Who's At The Keyboard? Authorship Attribution in Digital Evidence Investigations", International Journal of Digital Evidence, V4#1, 2005.

differentiate between subjects and the quality of the results they yield are not well understood in most cases, and only limited studies have been performed on these issues. As a result, using them in any definitive way is dubious, but that does not preclude their use as indicators if the examiner properly couches them in terms of their reliability and known characteristics. While there have not been a large number of studies done of these attribution techniques, to the extent that they provide insight or become part of a larger picture, they may reasonably be applied.

Using redundancy to build a consistent pattern

One of the approaches available is the use of multiple indicators of user behavior, presence, activities, and authentications in concert to build up a pattern of use that is consistent with the individual and inconsistent with other individuals in the available population of suspects. As more of these indicators are consistent and none are inconsistent, the effect is cumulative on providing probative information. While none of these things will produce a definitive attribution on their own, in combination, they may exceed the threshold of credibility required to be used in a legal matter, and they may survive challenges.[285] As more redundant indicators are used to attribute to individuals, forgery becomes more difficult, but may remain feasible for sophisticated threats.

Examples of approaches include identifying characteristics common of people of different sexes, education levels, training in particular areas, different first languages, and similar differentiators. These methods are potentially probative with regard to particularization even if not useful for individualization except in environments where these are effectively the same (e.g., a small number of people with particular mixes of these differentiators and the assumption of no deception being used).

Summary of human attribution from DFE

In any case, the examiner should be careful not to make leaps of faith in attribution. Rather, they should clearly state the traces and events that support an attribution and known error rates and different sorts of errors that may limit the probative value of results

285 F. Cohen, "Attribution of messages to sources in digital forensics cases", HICSS-43, Jan 7, 2010.

associated with the techniques applied. An examiner seeing such results without accurate error rates or detailed descriptions of the techniques used, should point out the limitations of these methods and the lack of scientific evidence for their utility in attribution.

Admissibility of testimony and evidence of the results of examination using these methods is largely undecided, and it is somewhat of a stretch to call them scientific, at least as they are commonly characterized today.

> The error rates are substantial for the methods for which error rates have been experimentally examined, and there have been too few studies to allow them to be well characterized even for limited populations under controlled circumstances.
>
> Some of the methods are subject to subversion in some malicious attack environments, and the overall situation in which they are applied must be considered in their use.
>
> Some of them require traces that are often unavailable or difficult to attain.
>
> Many content-based methods have shown poor reliability under deception, even with small populations, and are not demonstrated to be significantly better than chance.

While these limitations don't strictly preclude their use, it does lead to questions about the probative value of these methods as opposed to their prejudicial value, issues that ultimately have to be settled by the courts.

Attribution of actions to automated mechanisms

While attribution of actions to individuals, identified as "level 3 attribution in the discussions above, is problematic, attributions to the direct network source (level 1) and indirect network source (level 2) are more readily attained if enough of the appropriate sensors are in place in real-time and adequate methods are used. Attribution of actions to physical devices, computer software programs, and association with sequences of bits are also at issue in DFE examination.

Level 1 network attribution

From a forensic standpoint, level 1 and 2 network attribution with reasonable surety is difficult unless sensors were in place and properly recording during the activities in question. Nevertheless, level 1[286] and level 2[287] methods have been explored and are well summarized along with level 3 results.[288]

Level 1 attribution has been characterized as follows; "[Assume] an IP packet, P, is generated by a machine, G, forwarded by a sequence of IP routers, and finally, if not dropped along the way, delivered to a recipient machine. ... The goal of Level 1 attribution is, given P, to identify G." Link identification may be used to limit packet flooding by collecting data along the travel path and identifying which path sources the packets at step after step leading back to the originating source. Trace approaches require that intervening infrastructure add trace packets or traces to packets to allow them to be tracked. The IP addresses within the packet are most commonly used, but this can be readily circumvented by forgery by actors along the path to and from a source.

Cooperating parties may also tunnel packets through intervening infrastructure and track those activities. Logging methods and search of logs from intervening computers are also feasible when those records are generated, kept, and available. Remote monitoring sensors may be used if placed in advance, and ingress or route-based filtering can be used to limit and thus help to identify sources. They summarize their analysis in Table 7.2 below.

Active methods have been used for investigative purposes, such as the induction of datagrams into networks so as to alter traffic by reducing available attack bandwidth and, by so doing, detecting difference´s between sources and travel paths.[289] However such

286 D. Cohen and K. Narayanaswamy, "Techniques for Level 1 Attack Attribution", 2004/04/08, CS3, Inc., available at: http://isis.cs3-inc.com/level1-x.html

287 D. Cohen and K. Narayanaswamy, "Techniques for Level 2 Attack Attribution", 2004/03/22, CS3, Inc., available at: http://isis.cs3-inc.com/level2.html

288 Survey / Analysis of Levels I, II, and III Attack Attribution Techniques", available at: http://www.cs3-inc.com/arda-survey.pdf

289 Hal Burch and Bill Cheswick, “Tracing Anonymous Packets to Their Approximate Source”, Proceedings of the 14th Systems Administration

techniques are problematic for examination because, among other things, they require real-time activities by the investigator which the examiner then has to analyze for attribution, and they are based on assumptions about networks, “attack” behaviors (e.g., high volume ongoing forged datagrams), network behavioral characteristics, and many other things, that might not be true.

Method	*Trace one packet*	*Works with existing routers*	*Advance notice needed*	*Additional comms required*	*Other problems*
Link Identification methods					
Link Test	no -	yes +	yes -	varies	yes -
Itrace	no -	no -[1]	no +	no +	no +
PPM	no -	no -[1]	no +	no +	no +
DPM	yes +	no -[1]	no +	no +	no +[2]
Tunnel	yes +	yes +[3]	no +	no +	no +
SPIE	yes +	no -[4]	no +	yes -	yes -[5]
Monitors	yes +	yes +	yes -	yes -	no +
Packet Filtering methods					
Ingress Filtering	yes +	yes +	no +	no +	yes -[6]
Route Based Filtering	yes +	no -[7]	no +	no +	yes -[8]

Table 7.2 - Level 1 attribution technique summary[290]

They note that: "[1] This may be possible using (possibly nonstandard) router features to route a subset of packets to another cooperating machine. [2] Marking across non-cooperative infrastructure requires use of tunnels. [3] It is not clear whether tunneling will work with high speed routers. [4] Current work attempts to monitor links with a separate machine. Very high speed links would still require new special purpose hardware. [5] SPIE has to deal with the problem of trading off between additional memory

Conference, Dec 3-8, 2000 – New Orleans, LA, pp 319-327.

290 D. Cohen and K. Narayanaswamy, "Techniques for Level 1 Attack Attribution", 2004/04/08, CS3, Inc., available at: http://isis.cs3-inc.com/level1-x.html

vs. time. [6] Ingress filtering has a very poor effectiveness to degree of cooperation ratio. [7] Current routers are able to filter, and in some cases there will be few enough filtering rules to use this mechanism without unacceptable cost. However, we expect that will not always be the case. In the general case, filtering is similar in complexity to routing. [8] Route based filtering assumes that relevant routing data can be obtained."

From a standpoint of DFE examination, the traces associated with the relevant events must have been collected and preserved, and must be available to the examiner, and these techniques have not been shown to be uniformly effective or reliable in practice. They largely depend on cooperation from other parties, and if those parties either don't cooperate or act to defeat the attribution process, it will normally be circumventable.

Level 2 network attribution

According to previous work,[291] "the goal of 'Level 2 attribution' is to find the beginning of the 'causal chain' that leads to that activity. The result is again an activity along with the computer in which it occurs." It clarifies limitations that affect the forensic examiners situation: "The most obvious approach to finding the beginning of a causal chain is to trace back one step at a time from the activity to be attributed. Most of what we describe here is concerned with how to take that next step backward in the causal chain. ...There are no techniques that necessarily lead directly to the primary controlling host [and] taking one step back along the causal chain is not actually guaranteed to get 'closer' to the goal."

In order to be effective, such methods must undo the effects of anonymizers, network address translation (NAT) gateways, proxy servers, temporary address schemes, such as those provided by the dynamic host configuration protocol (DHCP),[292] and any of a myriad of other similar mechanisms.

291 D. Cohen and K. Narayanaswamy, "Techniques for Level 2 Attack Attribution", 2004/03/22, CS3, Inc., available at: http://isis.cs3-inc.com/level2.html

292 R. Droms et. al., "RFC 3315: Dynamic Host Configuration Protocol", available at: ftp://ftp.rfc-editor.org/in-notes/rfc3315.txt

Several identified problem cases include without limit, the "reflector", "stepping stone", "custom software", "zombie", and "physical control". Methods listed include "internal monitor", "logs", "snapshot", "network traffic", and "reactions to tracker or defender activity". Under the assumption that level 1 attribution works, which is problematic from the DFE examiner's perspective because they rarely have all of this information after the fact, Table 7.3 describes these techniques and their limitations.

In Table 7.3, a "+" indicates techniques that use the given data and work reliably given a machine controlled in a given way to find some upstream controlling machine, if there is one, is available. A "-" indicates the absence of such methods. A blank is used to indicate that level 1 attribution is sufficient.

source	*reflect*	*stepping stone*	*custom software*	*zombie/ physical*
monitor		+	- [1]	-
logs		+ [2]	+ [2]	+ [2][3]
snapshot		+ [4]	- [1][4]	- [1]
net traffic	[5]	+ [6]	-	-
reaction		+ [6]	- [7]	- [7]

Table 7.3 - Level 2 attribution summary table[293]

The following notes apply; "[1] This is useful for determining how a program controls behavior and for determining the source address of any ongoing outside control, but is not sufficient for attribution in the absence of controlling communication. [2] Existing logs are generally not sufficient but useful for at least narrowing the possibilities. Additional logging is possible and useful. [3] The logs must resist alteration, e.g., by being recorded on another machine or on write-once media. [4] This is as effective as internal monitor but higher cost. [5] The ability to observe network traffic is needed for Level 1 attribution. [6] This works in the absence of strong

293 D. Cohen and K. Narayanaswamy, "Techniques for Level 2 Attack Attribution", 2004/03/22, CS3, Inc., available at: http://isis.cs3-inc.com/level2.html

anonymization. [7] One problem is that the attacker cannot in general be forced to react. ..."

From a DFE examiner's perspective, only the entries with "+" are potentially helpful, which leaves only (1) internal monitoring, if present, (2) logs, which are typically insufficient for definitive results in all cases, but which may be available if sought, and must resist alteration, (3) snapshots, which are almost never available to the examiner, (4) network traffic details, which are only available in limited cases, and (5) reaction, which can work only for stepping stones and only when strong anonymization is not in place. While this does not leave the situation hopeless, it seems clear that logs are the most likely to succeed, tend to meet the normal business records requirements, are subject to subpoena in many cases, and tend to be more available than alternatives.

Network attribution caveat

To keep this review in context, level 1 and level 2 attribution as discussed here assume, except where noted, that a malicious attacker is trying to circumvent normal controls for their own ends. This is not always the case in DFE examination, but it should be considered as a possibility in all cases, particularly because of the high number of systems in the Internet today that are under attack and the large numbers of documented successful penetrations.

Device identification and attribution

A substantial amount of effort has been put into methods for hardware identification at the hardware level. This generally breaks down into physical methods that are beyond the scope of this work, and digital methods that are not.

Generally, devices may have brand identifiers, model numbers, serial numbers or other identifying digital indicators that are readable and sometimes temporarily or permanently alterable from software interfaces or within hardware devices. To the extent that traces are available to the examiner, this presents an opportunity for attribution of actions to devices. Computers are often compositions of other devices each of which may have identifying digital indicators. For example, Ethernet cards, processors, disk interfaces, external memory devices, SIM cards, WiFi cards, and other similar peripherals typically have these identifiers.

The identifying information from such devices is commonly collected as part of the system startup or hot swap mechanisms as part of their normal initialization process. This allows proper drivers to be put in place to interface to those devices. System logs, device logs, and other similar logs often keep track of all such activity and include the identifying information in the content they capture and preserve. This information is also sometimes available and referenced in registration or purchase information that is held by disinterested third parties, is sometimes transmitted as part of registration or execution processes, is sometimes checked in system startup and operation to allow configuration verification or alteration, and is sometimes contained in registries, configurations, document files, and in other similar locations. These activities often leave traces that become available to the examiner, and may be used to associated specific devices with specific activities. This assumes that the traces are accurate and unaltered and that the mechanisms that made those traces were operating properly and properly captured and recorded the original data.

To the extent that time or other related traces so indicate, and to the extent that the set of devices present at any given time can then be derived based on analysis and interpretation of traces and events, these may be leveraged to attribute sources of traces to devices with the level of certainty determined by the certainty of the component traces, analyses, and interpretations. This notion was introduced by Carrier[294] as part and parcel of his model for time analysis, but not applied specifically to the attribution problem.

Recent advanced in hardware embedded integrity controls have been applied to many areas, including both digital rights management, and operational integrity controls. Integrity controls based on the notions of using cryptographic checksums for integrity protection,[295] ultimately led to the integrity shell and related

294 B. Carrier, "A Hypothesis Based Approach to Digital Forensic Investigation." PhD Dissertation; Purdue University; May, 2006.

295 F. Cohen, "A Cryptographic Checksum for Integrity Protection", IFIP-TC11 "Computers and Security", V6#6 (Dec. 1987), pp 505-810. See: http://all.net/books/integ/checksum.html

approaches[296] and to high integrity boostrapping.[297] These have been combined in hardware devices through the efforts of the Trusted Computing Group (TCG) and their Trusted Platform Modules (TPM) and are now integrated into millions of computers in use and production all over the world.[298]

With these devices used to control execution and use in computers, and assuming they keep audit and related information in their trusted storage or authenticate those traces in untrusted storage using trusted storage and device mechanisms, a high degree of certainty can be attained in the accuracy of authenticated audit records. In addition, the secure bootstrapping processes that some such mechanisms implement allows the specific devices in use to be authenticated at the hardware level. This then leads to identification of devices and authentication of related device information that may be leveraged to attribute actions to devices in a far more certain manner than can otherwise be attained from traces in other environments.

Operating environment identification and attribution

Behavioral indicators are often also available for the operating environments executing as FSMs within devices. As an example, the typical output patterns and interactions between specific versions of specific operating systems have been characterized at the network traffic level based on behaviors of Internet Protocol (IP) stacks.[299] This approach has been applied in various software mechanisms to identify operating systems from IP traffic, but it is easily deceived, and such deception can be carried out by relatively unskilled attackers using widely published methods. More skilled

296 F. Cohen, "Models of Practical Defenses Against Computer Viruses", IFIP-TC11, "Computers and Security", V7#6, December, 1988. Available at: http://all.net/books/integ/vmodels.html

297 F. Cohen, "A Note On High Integrity PC Bootstrapping", IFIP-TC11 "Computers and Security", V10#6, October 1991. Available at: http://all.net/books/integ/bootstrap.html

298 See: https://www.trustedcomputinggroup.org/home for details on the current status of this effort.

299 R. Lippmann, D. Fried, K. Piwowarski, and W. Streilein "Passive Operating System Identification From TCP/IP Packet Headers", July 2004, Massachusetts Institute of Technology, See: http://www.ll.mit.edu/mission/communications/ist/publications/03_POSI_Lippmann.pdf

attackers have demonstrated the ability to create intentional appearances of other systems that will be misidentified by such methods. Perfect emulations of networked systems are nearly attainable under some circumstances,[300] and more sophisticated approaches to deception allow multiple systems and operating environments to be emulated with relatively simple devices.[301]

Within an operating environment, programs and their behaviors also produce operating environment-specific traces. For example, system calls within different environments produce different results in terms of error codes, messages in log files, pathnames, device names, time and space granularity, and so forth. These may also be used as indicators to allow attribution of an operating environment to traces, and in many cases, serial numbers or other similar components may be present in some of these traces. This approach has been applied for consistency checks in various operating environments, but this too is easily deceived.[302,303] Similar methods include virtualization approaches, which can operate one operating environment embedded within another, and simulation, which can simulate one environment within another to the level of accuracy desired. Simulation methods and emulators for hardware devices that operate within other operating environments have been available for a long time, while modified operating system calls for deception are more recent. Each produces problems for asserting with high certainty the attribution of a trace to an operating environment.

While it may reasonably be argued that an attribution is correct in the emulated or simulated environment in that the attribution to the environment is accurate, regardless of what outside environment the operating system is operating within, the same cannot be said

300 F. Cohen, I. Marin, J. Sappington, C. Stewart, and E. Thomas, "Red Teaming Experiments with Deception Technologies", 2001, available at: http://all.net/journal/deception/RedTeamingExperiments.pdf

301 F. Cohen, US Patent 7107347 - "Method and apparatus for network deception/emulation".

302 F. Cohen, “Method and Apparatus for Providing Deception and/or Altered Execution of Logic in an Information System US Pat. 7,296,274.

303 F. Cohen, D Koike, V. Nagaeu, “Method and Apparatus Providing Deception and/or Altered Operation in an Information System Operating System”, US Pat. 7,437,766, and others in these families.

for the more recent deception environments. The ability to create such an operating environment and emulate whatever behaviors are desired puts into question any attempt to tie the traces from the operating environment into the hardware devices being used. Clearly, the virtualized environment may contain a virtualized version of a hardware environment, except to the extent that the virtualized environment does not have access to the internal protected keys of the trusted platform module (TPM)[304] in such a system, but rather only limited use of those keys through the external interface to the TPM. Care obviously must be taken in making such attributions and in the way in which they are put forth by the DFE examiner.

Complexity-related authenticators

The cryptographic methodologies used in TPMs and in previous works on the use of cryptographic checksums[305,306,307] along with the digital signature work associated with the RSA cryptosystem[308] and previous related work on the concept of digital signatures[309] leads to a reasonable basis for asserting mathematical properties that, in conjunction with the physical mechanisms of a TPM or other physical measures, provide a reasonable basis for asserting, with a defined degree of certainty, that a consistency between content and a digital signature found in a trace, indicates that the signature of that trace associated with the content in the trace being signed was undertaken by a mechanism that had access to the private key used in signing. From this, it may be argued, for example, that

304 https://www.trustedcomputinggroup.org/home for details.

305 F. Cohen, "A Cryptographic Checksum for Integrity Protection", IFIP-TC11 "Computers and Security", V6#6 (Dec. 1987), pp 505-810. See: http://all.net/books/integ/checksum.html

306 F. Cohen, "Models of Practical Defenses Against Computer Viruses", IFIP-TC11, "Computers and Security", V7#6, December, 1988. Available at: http://all.net/books/integ/vmodels.html

307 F. Cohen, "A Note On High Integrity PC Bootstrapping", IFIP-TC11 "Computers and Security", V10#6, October 1991. Available at: http://all.net/books/integ/bootstrap.html

308 R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public Key Cryptosystems", CACM V21#2 (Feb. 1978) PP120-126.

309 Diffie, W., and Hellman, M. New directions in cryptography. IEEE Trans. Inform. Theory IT-22, (Nov. 1976), 644-654.

given that the private key used was always generated and stored within a TPM or other physically secure device, that the physical device was used in the signature.

This is fairly stunning given that it is entirely due to traces that need not otherwise be authenticated. But unless definitive possession of the device or other physical limits can be shown, and unless device use can be attributed to a particular individual, attribution may be limited in certainty to the device. Such methods are also limited by the nature of the method used for the actual signing. For example, if the signature was the result of a process that first compressed the original content into a smaller message digest, the entire signature is subject to the limits of the message digest process. But this process may itself be flawed.

An example of such a flaw is the ability to forge, although not consistent with normal syntactic analysis, the MD5 message digest of content, by adding a homing sequence of the MD5 FSM to the beginning of the sequence being digested.[310,311] This homing sequence will put the MD5 mechanism into the initial state after processing some sequence of bytes, and from that state, subsequent action of the FSM is identical to the processing without the homing sequence. In a more general sense, a homing sequence of a more general type may be placed within a sequence being digested, presumably with any message digest technique, and it will return the content to a state from which original content may be digested, producing an identical message digest to the original sequence, but with a different sequence added.

This entire class of attacks and failure modes is all a result of the information physics problems that digital space converges and homing sequences exist for certain classes of FSMs. It is guaranteed that such mechanisms exist for essentially all such digest mechanisms. In addition, there are often partial homing sequences that exist for other FSMs such that, once in a particular state, the machine may be able to be returned to that state by putting some set of conditions on subsequent inputs. If those

310 X. Wang, D. Feng2, X. Lai, and H. Yu, "Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD", http://eprint.iacr.org/2004/199.pdf

311 M. Steven, "Fast Collision Attack on MD5", available at the URL http://www.win.tue.nl/hashclash/

conditions can be satisfied while putting in meaningful content with compatible syntax, then the digest can be defeated in a more meaningful manner.

The question of meaningful alteration is another important thing to note. The DFE examiner seeking to use digests or other similar mechanisms for attribution may perform an analysis that identifies that the content matches the digest formed with a key controlled in some manner, and also determine that the results meet the syntax requirements of the type associated with the trace. If the type has inherent redundancy, such as the syntax of an HTML page or text from a letter or document, the assertion that the result may be a forgery becomes far harder to show. The reason is that it would have to be shown that a homing sequence exists that meets the format requirements and is of a length that would fit within the traces identified.

In some cases, it may even be possible to exhaust the space of such homing sequences, or show that the digest comes to a different result for each trace in which one or more sub-traces are removed from the trace. This would then show that known or published methods by which forgeries can be accomplished were examined and found not to exist for the particular trace under consideration. Such an analysis is surprisingly easy for most current message digest systems. But the examiner should be careful not to overstate such a case. While these methods may rehabilitate evidence associated with such complexity-based mechanisms, they are all inherently imperfect, and it is usually possible that such a trace, even though consistent, would be false.

Public key infrastructure is also used to support most current signature methodologies, and as such, the infrastructure must be properly functioning in order for any attribution associated with that infrastructure to be effective. This is not limited to the availability of the infrastructure in order to look up attribution information. The integrity of the infrastructure is also key to its utility for attribution, as is control over its use and accountability associated with it.

Recent attack methodologies have shown reliable ways to forge the message digests used to authenticate digital certificates, resulting in the ability to forge Web site certificates well enough to deceive

most Web browsers.[312] The computers holding these keys may be subject to attack, name space problems may be exploited to make things appear different than they are, and a wide array of other similar problems at the infrastructure level may defeat attempts to accurately attribute based on materials associated with these systems.[313] The originating identities upon which such mechanisms are based are also subject to attack, errors and omissions may result in incorrect attributions, and getting witnesses to testify about such matters may be expensive and problematic.

Increasing instances have been publicized in the media of the use of forgeries of public key infrastructure (PKI), and in addition to external malicious attack on such infrastructures, it was revealed in the press that trusted PKI providers have provided back doors to support false authentication to governments and customers to allow them to forge identities and authentications systematically.

Predicted behavior of programs

Computer programs are finite state machines, but they tend to have large numbers of states, and are typically implemented without the same level of attention to assurance as the hardware devices they operate within. The anomalies associated with their operation and their normal operating characteristics may be used to attribute actions to the software executed.

A typical attribution process starts with type analysis to determine the type of the trace and associate it with a known format or detect consistency with events. Programs typically behave within an envelope of behaviors of similar classes of programs. For example, programs that process Web requests over a network typically handle queries using the hypertext transfer protocol (HTTP)[314] and send results using a version of hypertext markup language (HTML)[315] or other formats associated with the underlying protocols and languages. Similarly, databases typically handle requests in a

312 K. Poulsen, "Researchers Use PlayStation Cluster to Forge a Web Skeleton Key", Dec 30, 2008, see http://blog.wired.com/27bstroke6/2008/12/berlin.html

313 F. Cohen, "50 Ways to Defeat Your PKI and Other Cryptosystems", 1999 annual CSI conference in Washington DC. http://all.net/journal/50/crypt.html

314 R. Fielding, "Hypertext transfer protocol", RFC 2616, 1999 defines HTTP version 1.1 and other RFCs define other versions thereof. See: http://www.ietf.org/rfc/rfc2616.txt

database language such as the Structured Query Language (SQL),[316] email typically gets sent to computers using the simple mail transfer protocol (SMTP),[317] and Joint Photographic Experts Group (JPEG) graphical files[318] have a header that indicates the version of JPEG, extensions, and application-specific information.

Once a language or protocol is identified, the programs processing it may be identified by the examiner. For example, the contents of email messages may have the name of the servers embedded in their "Received:" headers, while a graphical image file (GIF) may have sourcing information contained within the header area, and the response from a Web server exchange may indicate the name of the server being used. Attribution may be furthered as simply as by looking at the traces and noticing the string associated with the mechanism. For example, MySQL, PostgreSQL, Oracle, mSQL, and Microsoft SQL are all identified with SQL variations,[319] and there are hundreds or more Web servers in use, many of which provide indicators including the server name, version, and other related information in their replies to HTTP requests.

If such a program is indeed present in the available traces and if the startup mechanisms of the operating environment as indicated by traces shows that such a program would normally be used to handle such exchanges, then this is consistent with the attribution. If events support this interpretation, the attribution becomes confirmed still further.

Specific versions of specific programs can be tested to identify the traces they normally leave, and this will be discussed further in Chapter 8. From an attribution perspective, the processes of

315 The World Wide Web Consortium at http://www.w3.org/ defined the various XML specifications and related protocols including identification of versions and specifics relating to placement of names of servers, clients, and other related matters.

316 SQL is described and defined at http://www.sql.org/ complete with descriptions of variations on SQL used in different software.

317 J. Postel, "Simple Mail Transfer Protocol", RFC 821, 1982 defined the SMTP protocol at http://www.ietf.org/rfc/rfc821.txt and subsequent versions are reflected in other RFCs.

318 E. Hamilton, "JPEG File Interchange Format Version 1.02", 9/1/1992, http://www.jpeg.org/public/jfif.pdf

319 http://www.sql.org/

analysis and reconstruction can be used to gain increasing confirmation that traces are consistent with events. Consistency demonstrates the potential validity of indicators used to attribute traces to programs that cause them. Programs also tend to leave traces in the form of log files, they may leave traces in specific directory areas within specific operating environments, they may have process identities indicated by traces, and each of these may have metadata in the form of identifying numbers, date and time stamps, and so forth. Again, analysis of trace consistency may be used to strengthen the attribution of traces to programs.

Deceptions are also used in some Web servers, mail servers, and elsewhere, with the object of reducing intelligence activities that could be used for attack, to mimic properties of other mechanisms so as to be treated as other applications are treated, to provide anonymity, or for other reasons. For example, special properties of specific servers are used by many Web browsers, and properties of browsers are used by many Web servers, to determine how to respond to messages. Software that follows specifications laid out by other programs but does not have the market clout to get special attention by the makers of more popular servers or browsers, may use these deception mechanisms to provide those features in a manner that is transparent to other parties in transactions. Such mechanisms are common, but this does not prevent the use of these indicators to confirm or refute trace consistency in cases where a given browser, server, or other software component is known to be present and being attributed to a trace. The examiner should be cautious to consider the potential for deception in this context, as its use is widespread and often considered legitimate in these sorts of situations.

Limits of attribution to automated mechanisms

The problems with attribution of actions to automated mechanisms are substantial, but on the other hand, there are some substantial advantages from information physics that may help with attribution. While the high level of predictability of digital systems helps make crisp distinctions between consistent and inconsistent traces and events, it also makes such systems susceptible to precise forgery.

While most systems have predictable behavior when all is well, attacks are frequent and breakins common, putting traces at risk of containing indicators of the attacks that introduce inconsistencies with traces from normal operation. Because attacks tend to bypass normal controls, the normal limits that operating environments place on the extent of traces produced by any program become ineffective, and traces may extend beyond their normal envelope. Because attack detection is, in general, undecidable, and in practice imprecise,[320] in most environments, the examiner cannot rule out the possibility that such attacks have taken place and have produced altered traces.

The best that can be said in most cases is that; after examination which sought known corruptions and attacks, and indicators of unknown attacks; no inconsistencies were found that would tend to indicate the presence of such methods. The traces appear to be consistent with the attribution results produced. This, of course, assumes that no such indicators are detected and that they are sought. As an alternative, in cases where known attack mechanisms are identified, it may be feasible to limit the scope of their impacts on traces by attributing behaviors to those known attack mechanisms in the same way as attribution of actions to other programs are done.

In digital forensics, and in particular in examination, altering the situation that exists is legally problematic. While such techniques may be applied in attack attribution,[321] the DFE examiner should be aware of the dubious legal standing of such methods.

Information physics attribution limits and approaches

Other aspects of information physics may allow or limit certain types of attributions. Examples are identified here for some of the less explored issues in attribution.

320 F. Cohen, "National Info-Sec Technical Baseline: Intrusion Detection and Response", Lawrence Livermore National Laboratory and Sandia National Laboratories, December, 1996. see: http://all.net/journal/ntb/ids.html

321 D. Wheeler and G. Larsen, "Techniques for Cyber Attack Attribution", Institute for Defense Analysis, IDA Paper P-3794.

- Finite time granularity (the clock) may limit the accuracy with which traces may be said to be consistent or inconsistent with attributions.
- Exact copies imply that it is impossible to tell whether an attribution is from an original source or a copy of some other source.
- Theft without direct loss implies that attribution is not harmed in the direct sense when copies of traces are taken, so when attack or other mechanisms that only read content are at issue, they do not degrade the quality of the traces. Many attack mechanisms are read-only and do not hinder attribution associated with some traces, even though they may create problems for uniqueness based assertions about traces. Forensically sound "copies" are as good as originals.
- Finite (fast) rate of movement implies that time can be taken into account in attribution and that causality must be reconciled with delay times for processes. This is often helpful in refuting claims of causality when, for example, there is not enough time available for the asserted causes to produce the asserted effects.
- Finite State Machines (FSMs) and their properties apply to attribution based on DFE and there are a wide range of results that apply here on a case by case basis.
- Homing sequence implications have been discussed for message digests, but similar limitations exist in other sorts of attribution where state effects sequence.
- Forward time perfect prediction is helpful in attribution based on reconstruction, which will be discussed in Chapter 8.
- Backward time non-uniqueness is really a core problem for attribution since it means there may be many causes for any particular trace. Attribution generally seeks to restrict the envelope of possible causes for a trace, and to the extent that reverse time can be constrained by other traces, events, or analysis, this will help to mitigate this problem.

- Digital space converges in time, and for attribution, this has many of the same consequences as non-uniqueness of reverse time.
- The results are always bits, which implies that there are limits to uniqueness and that there is only so much depth to which the examiner, or must, can go in attribution.
- Results are always "Exact", which means that unexplained inconsistency is a big potential problem. Something must have caused the inconsistency, and if it cannot be explained, traces and attributions cannot reasonably be authenticated.
- Time is a partial ordering, which implies that causality often cannot be definitively established, because a definitive causal relationship demands that causes precede effects.
- Representation limits accuracy, so attribution will have finite accuracy.
- Precision may exceed accuracy, and the examiner must be careful to use the proper level of precision in characterizing results related to attribution.
- Forgery can be perfect, which means that attribution cannot be perfect.
- DFE is almost always latent, which means that the mechanisms by which attribution are done must be understood through tools and the tools are also subject to imperfections and must meet surety requirements. Most of the methods that tools are based on are limited in terms of their accuracy, and care must be taken by the examiner not to believe the higher accuracy presented by tools than the method on which they are based.
- DFE is trace but not transfer. Thus no part of one thing is left with the other, and bits are fungible rather than particular.
- DFE is circumstantial, which means that attribution is necessarily circumstantial as well.
- DFE is hearsay and attribution is therefore subject to hearsay limits on the admissibility of the traces it is based upon. Since attribution often involves fusing together traces

from many different sources, the hearsay problem may be more complicated and involve more elements than for other trace analysis and interpretation.

- DFE alone cannot place a person at a place at a time. To the extent that attribution seeks to do this, it has limited value. These limits should be clearly understood and explained so that attribution can be meaningfully understood in context.
- DFE can show consistency or inconsistency only, so the interpretation of attribution should be kept in those terms.
- Probability is dubious in most efforts related to DFE. It is often easy to fall into the mistake of trying to use probability to bolster attribution when there are multiple attributing traces. The use of Bayes theorem for multiple attribution methods, statistical independence assumptions, or stochastic process assumptions, are problematic at best.
- Content has information density that is variable by nature, and this nature may be leveraged for attribution to the extent that it is meaningful. But historically, this is only of limited value. For example, language characteristics have density properties that are useful in certain types of cryptography, but this will be of little help in attribution, because the language is normally determined with ease. The use of these characteristics for classifying writers or mechanisms has not been highly successful to date.
- Digital signatures, fingerprints, etc. generated from content have been discussed at length and have limits and benefits identified earlier.
- Content meaning is dictated by context, and from an attribution standpoint, this implies that the context used to determine the attribution must be well understood and reasonably explained. The problems with virtualization and emulation as well as other deception-related issues demonstrate the need to establish context, and the examiner must exercise care in characterizing context.
- Context tends to be global and dramatically changes meaning. This means that, in cases where there are global

contextual issues, different contexts may have a dramatic effect on attribution. While there isn't always another layer of context that could dramatically alter the interpretation of attribution, there may be. For example, there may be a trace indicating a bluetooth keyboard that could have remotely controlled the computer in question. That might completely alter the attribution of traces to individuals at keyboards.

- Cognitive limits of programs may produce outputs that are completely senseless when examined in detail. For example, if an attribution analysis program derives 247 features that attribute actions to an actor, and those 247 features cannot be explained sensibly by a person to a person, they are not likely to be meaningful to a jury if challenged. They may be a result of samples provided to the program or the order in which samples were provided, and not actually related to meaningful features that are valid for attribution. They may produce completely wrong answers on the next sample.
- Time limits on achievable results limit the ability to try attribution methods, and many of them may take a lot of time and effort. There is no standard library of these methods and devising and implementing software for this purpose is likely to be error prone and time consuming as the number of methods increases.
- Time and space tradeoffs may allow determinations to be made that particular causes could or could not lead to particular effects with the available time and space.
- Near perfect virtualization and simulation are possible, which leads to the problems identified above that limit the perfection of attribution.
- Undecidable problems mean, among other things, that it cannot be definitively shown for most cases that the traces are not the result of a subversion of some sort, make it practically impossible to definitively show that the traces resulted from a system that was operating properly, and otherwise limit the best case for attribution.

- Computational complexity limits computations, and this is the basis for the analysis of signature based methods for attribution. It also acts as a speed-of-light for analysis.
- Complexity-based designs are the basis for the authentication methods based on public key cryptography.
- Consistency is guaranteed, which is to say that inconsistency leads to refutation. However, in the case of the attribution methods discussed here, the methods are flawed, so limited inconsistency does not necessarily rule them out any more than consistency makes them definitive.
- Hardware fault models may lead to understanding of limits on trace consistency, particularly when very small numbers of inconsistencies are found over a controlled envelope of space and time in large complex environments with many computers operating over signifiant time frames.
- Accidental assumption violations are likely when made about the methods discussed herein. Such assumptions should be scrutinized with care before use.
- Intentional assumption violations through malicious attack are likely to be problematic if there are indications of such things, or if it cannot be shown that the systems are free of these things to a reasonable degree of certainty. The threat environment is key to addressing this issue.
- Discontinuous space and time are likely to be problems if the accuracy required for the technique being applied exceeds the available granularity. For example, because file system granularity is typically on the order of seconds and keystroke timing is on the order of milliseconds, the relationship between these may be problematic. Similarly, jitter in networks may be sufficient to invalidate any results related to keystroke timings over distant networks for attribution purposes. This also relates to the issues of discontinuous time, and the amplification of minor differences and suppression of major differences near and far from discontinuities, respectively.

- Identical use of an interface may produce different results, leading to errors in methods using physical interaction with computer systems. This includes all biometric methods. In particular, variances in inputs forces distance to separate inputs, forcing tradeoffs between false acceptance and false rejection. This impacts the relation between the size of the space and the meaningfully distinguishable population size.
- Ordering and value sort reversal may also disrupt analysis methods used in attribution, and sensor and actuator limits in terms of physical properties, clearly limits the utility of biometric interfaces. Clearly, some Δ must be included in all such analysis. This then limits accuracy and precision and the number of differentiable individuals.

Making assumptions to make progress

In the area of attribution, making assumptions may help to bring progress. But the scientific basis for many of these methods is limited, and adding assumptions is likely to make results even more problematic than they already are. It is also important to consider that the other side might make different assumptions and that these may result in different attributions and causal relations.

Attribution of damages to parties

In civil matters, the issue of damages is often key to the issues in the case, and in criminal matters, demonstration of some level of damages may be necessary in order to trigger violations of laws. Even given that damages can be defined and characterized, those damages have to somehow be attributed to the parties involved.

NOLO Press,[322] defines damages as "money awarded to one party based on injury or loss caused by the other. ... different types or categories of damages [are]: compensatory damages ... general damages ... nominal damages ... punitive damages ... special damages ... statutory damages... treble damages". Drilling down:

> **compensatory damages** Damages that cover actual injury or economic loss. Compensatory damages are intended to

322 Online definitions URL: http://www.nolo.com/definition.cfm/term/A50A9EFC-8E6F-4B16-ABCAD9C0DD51CDEF

put the injured party in the position they were in prior to the injury. They are also called "actual damages."

This discussion is about attributing compensatory damages, a.k.a. actual damages, or simply "damages", to causes within a computer system. It is assumed for the moment that:

1. Any actual damages should be attributed to specific causes based on DFE,
2. Multiple causes are typically and simultaneously present for damages in real information environments, and
3. Attribution should allow damages to be proportioned based on the contribution of causes to the consequences.

This discussion ignores the rest of the attribution problem.

Actual injury or economic loss may be direct or indirect, and may be associated with seemingly unrelated phenomena, such as the slowdown in a computer causing a response to a request for proposals to be missed when it might otherwise have made the deadline. While it may be argued that it is the responsibility of a bidder not to run so close to deadlines, such issues will be ignored here in favor of the technical questions associated with (1) the ability to identify that there are actual damages and (2) the attribution of those damages to causes.

Summary of the legal environment

Recent legal rulings provide some guidance to understanding the technical issues. A recent California appeals case relating to emails sent to a corporation (Plaintiff) by Defendant will help understand the issues.[323] Ignoring the details of the case in favor of outlining the legal analysis, most of the issues at hand are not specific to a particular state, and the technical evidentiary issues appear to apply regardless. Finally, only issues related to damages associated with trespass are discussed here. To quote: "the contents of a telephone communication may cause a variety of injuries and may be the basis for a variety of tort actions (e.g., defamation, intentional infliction of emotional distress, invasion of privacy), but the injuries are not to an interest in property, much less real property, and the appropriate tort is not trespass." Nor are they of interest to the issues here.

323 Intel Corporation, Plaintiff and Respondent, v. Kourosh Kenneth Hamidi, Defendant and Appellant. No. S103781. Supreme Court of California

To summarize, "the general rule that a trespass to chattels is not actionable if it does not involve actual or threatened injury to the personal property or to the possessor's legally protected interest in the personal property". Damages are only relevant in the sense of the tort of trespass, which is the typical issue at hand in these sorts of matters, and trespass leads to damages only of the following five types:

1. **Physical Damage:** There are physical damages to the computer system. *This is almost never the case in DFE examination.*
2. **Conversion:** The computer system was no longer usable at all by its possessor. *This rarely occurs as long as the possessor has physical control and can rebuild the system for some useful purpose. Some attacks can result in the need for physical repair, like replacement of the BIOS chip.*
3. **Deprivation:** The possessor was significantly deprived of use to the point where the basic function of the computer was obstructed or completely lost. *This results from malicious attacks, when software fails, from configuration errors, and from many other causes.*
4. **Lost value:** The chattel lost some value, quality, or its physical condition was harmed, but this does not include the mere alteration of content where that does not deprive the possessor of use. *This may include leakage of confidential information, alterations that cause the system or applications not to function, and many other similar things.*
5. **Lost rights:** The possessor was deprived of some other legally protected interest such as a copyright, patent, or other interest or right. *Trade secrets disclosed might be an example of this.*

In addition, damages must be:

1. **Quantified:** The damages must be reasonably quantified by measurements taken. *This means that the examiner must be*

able to measure something from traces and events that allows the quantitative value of damages to be determined.

2. **Time framed:** The possessor must be deprived for a substantial period of time. *The examiner must be able to identify the time frame over which the deprivation took place and it must be substantial relative to some standard.*
3. **Tangible:** Damages must be the result of tangible trespasses and not merely the result of intangible ones, like electromagnetic emanations that do not deprive use. *The examiner must be able to show that the trespass occurred based on traces and events that demonstrate effects on the chattels owned by the possessor.*
4. **Unmitigatable:** The possessor must reasonably act to mitigate harm. *The examiner should be able to show that diligent efforts were applied to mitigate the harm by examination and analysis of changes to the system.*
5. **Uninvited:** The recipient must not invite the harm if they are going to claim damages. For example, if the harm comes from signing up to a service that is provided, the use of resources by the service is not actionable. *Traces may be consistent or inconsistent with this assertion.*
6. **Causal:** The damages must be proximately caused by the other party. *The examiner must be able to show proximate causality at some level of certainty by consistency of causality with the traces and events.*

These results are largely applicable for most or all of the United States and similar to the results likely in English and other related systems of laws. They will be described herein as damage types: {Physical Damage, Conversion, Deprivation, Lost Value, and Lost Rights} and forensically demonstrable properties: {Quantified, Time framed, Tangible, Unmitigatable, Uninvited, and Causal}.

Normally, the examiner is called upon to identify potential types of damage, methods by which damages may be attributed to causes, and methods by which damages may be measured and shown to meet the demonstrable properties. The examiner should also

provide attributions not available from events and show consistency of traces and events in the examination process.

Summary of the technical environment

From the beginning of timesharing, accounting has been a feature of computer systems. When computing resources were expensive, users were tracked and charged at different rates for disk usage, central processing unit (CPU) time, memory usage, and programs run were tracked and accounted for in this manner.

These sorts of records were historically used in conjunction with accounting mechanisms to charge customers for usage based on formulas and agreements. For example, published rate structures for systems and resources were applied against collected account and audit records to calculate bills and invoice customers. When usage exceeded prearranged maxima, the system would typically prevent further use until the next period or until the restriction was lifted, or apply another rate chart for overage. Similar restrictions are in place in systems today, for example in cellular telephone networks and other analog and digital networks, and in services such as mailing list servers, social networking sites, and other fee for services systems.

This sort of accounting remains operational on most Unix and Unix-like systems, mainframe platforms, Windows systems, and other systems still today, even if it is not as commonly applied. For example, the Unix "acct" command can be used to turn accounting on and off, and accounting is commonly enabled at system initialization. Even without this sort of accounting enabled, system, server, and program logs are generated and maintained as part of normal operation to allow for debugging and other sorts of review. Many modern and historic computer systems create and maintain audit records and metadata of various sorts. There are often file date and time stamp records, recorded logs of program executions, traces within output files associated with these and other programs, network-related records, and a wide range of other traces and records that may be present and that can be used to validate and make determinations about times, sizes, and related matters.

To the extent that metadata and audit records constitute normal business records available in traces, they can reasonably be relied

upon for integrity and accuracy, unless substantial inconsistencies are identified by the examiner. There is also a requirement that such records be retained when there is a reason to believe that they may be at issue in a legal matter, and therefore, such records should either be available, or the lack of those records should be used to produce a finding and to invoke sanctions.[324]

Overview of a damages attribution process

In attributing damages to parties, it may be helpful to create a table identifying the issues. Table 7.4 is a form for a simple analysis of the issues identified above as it applies to a case involving unsolicited commercial email (UCE) purported to be in violation of the US CAN-SPAM Act.[325] Similar tables can be created for other situations and a similar analysis will yield different specific results for each case.

Suppose an Internet Service Provider (ISP) claims damages from 100,000 emails of approximate average size 10,000 bytes, all of which had to be stored during delivery to clients, and all sent over a 30-day period. Statutory damages are $1,000 per email/recipient pair. This is a preliminary overview that an examiner might perform to get things started.

Issue	Example
Physical	This is not likely the case for ISPs processing emails.
Conversion	This is almost certainly not true as the systems could delete all of the emails and continue to operate.

324 The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production, September 2004 Public Comment Draft.

325 The "Controlling the Assault of Non-Solicited Pornography and Marketing (CAN-SPAM) Act", US Public Law 108–187—Dec. 16, 2003 - described at http://www.ftc.gov/bcp/edu/pubs/business/ecommerce/bus61.shtm by the Federal Trade Commission.

Issue	Example
Deprivation	10,000 bytes * 100,000 emails = 1 gigabyte. Most modern computers can store far more than this. Over a period of 30 days, this comes to an average of 38 bytes/second. Most ISPs sustain traffic of millions of bits per second, so this is less than 0.1% of bandwidth. This seems a specious claim. Handling emails does not take excessive amounts of time in normal operation, and certainly 100,000 emails are well within the normal limits of even an inexpensive computer system using a free email server.
Lost value	This does not appear to apply to this situation.
Lost rights	This does not appear to apply to this situation.
Quantified	Damages can only be established if there were measurements taken of performance of normal operations and it can be shown through those measurements that the usefulness of systems were significantly impacted by the emails in question.
Time framed	Damages are only demonstrable if specific amounts of time could be identified through the measurement process and tracked to specific periods. The 30-day period asserted appears to meet this criterion.
Tangible	This would apply to the deprivation of use case.
Unmitigat-able	The ISP probably could not sustain these claims unless they could show that the emails were bound for legitimate email accounts so that the emails could not be turned down out of hand, that they were not readily identifiable as UCE through some process that could reasonably be put in place, and that the sender refused requests to remove users from their lists.

Issue	Example
Uninvited	The ISP would have to show that the emails were not invited, for example, by showing that the emails were not requested by their customers and that the ISP didn't sell mailing lists for solicitation purposes or do some other similar activity. Another form of invitation might be an agreement with a third party to accept and store these emails, or a contract with the sender to transmit emails from them without limitations on volume or other similar constraints, or within defined service levels that are not exceeded.
Causal	The measurement would have to demonstrate that the emails in question were in fact causal with respect to the damages asserted and that the identified other party was responsible for sending the emails under the law in question.

Table 7.4 - An example analysis of preliminary damages evaluation

In Table 7.4, damage types {Physical Damage, Conversion, Deprivation, Lost Value, Lost Rights} and forensically demonstrable properties: {Quantified, Time framed, Tangible, Unmitigatable, Uninvited, and Causal} are listed along with a simplistic analysis. For damages to be reasonably claimed, they must be shown to be of at least one of the identified damage types and the claimant must demonstrate all of the forensically demonstrable properties. Similarly, to defend against such a claim, all of the claimed damage types must be addressed or any of the demonstrable properties must be successfully challenged. However; for legal matters, it is generally better to show or challenge all relevant damage types and forensically demonstrable properties.

Legal analysis also often involves more than a single thread of argument. A Plaintiff might assert multiple damage types and a defendant might present a number of defenses against these different claims. Indeed there is no need for the claims or defenses to be internally consistent or interdependent. Thus there may be multiple ways of measuring the same phenomena and they may yield different results, each of which might be presented by one

side or the other in an actual legal matter. As a result, a typical legal position may involve multiple instances of different types of damages and/or multiple different defenses for each and every claim and element of a claim. It is the role of the examiner to explore any and all of these issues to the extent that the available traces, events, and schedule allow this to be done.

A general approach to listing damages

Legal damage claims require substantial damage of defined types and forensically demonstrable properties. These damages are almost certain to be tied to some resource or resources within a computer or network because these resources are the very things that provide services the possessor may be deprived of or contain the content that could deprive the possessor of value or rights.

Internal resources within most relevant devices include; obvious ones, like disk space, memory space, and CPU cycles; and less obvious ones, like available file handles, process identifiers, protocol ports, and other similar kernel-level or programmatic resources that have limitations. Exhaustion of resources in these areas may produce inadequate space or time to complete a function within required service levels. External resources today are dominated by network bandwidth and, in some cases, input and output (I/O) bandwidth associated with disk drives or network storage arrays. While other input and output may also be involved, for the purposes of this discussion, these will be ignored, since they rarely have anything meaningful to do with the sorts of legal matters at issue. Power consumption, air conditioning usage, and other similar external resources may also be of import when they can be identified and linked to causes, and when their costs change as a function of the causes of import to the legal matter. Wear and tear that results in costs and maintenance that increases costs are also damages that may be identified in some cases.

Resource & type	Damage	Amount	Basis in properties
Disk space			
Memory (RAM) space			
CPU cycles			

Resource & type	Damage	Amount	Basis in properties
Network bandwidth			
Power, light, heat, etc.			
Wear and tear			
Maintenance, operational, and support costs			
Other internal resources			
Other external resources			
Indirect effects			

Table 7.5 - An example table of damages

It may be useful to create a table for the purposes of analysis in which the different resources and their associated actual injury or economic costs are listed and details of the causes are provided. Table 7.5 provides a basic format for this. It shows four columns. The first column is the resource type and actual resource (the actual resources have not been filled in within this example). The second column is the damage type experienced, which is one of {Physical Damage, Conversion, Deprivation, Lost Value, Lost Rights}. The amount of damages claimed follows in column 3, and is calculated by some method based on forensic data and analysis. The basis in Column 4 describes the basis for proof for the forensically demonstrable properties: {Quantified, Time framed, Tangible, Unmitigatable, Uninvited, and Causal}.

Filling in Table 7.5 table for different situations is a basic approach to damage assessment and attribution, and the results of such efforts along with the relevant detailed analysis, interpretation, attribution, traces, and events, will provide the foundation for presenting damages in legal matters.

Demonstrating the forensically demonstrable properties

When damages are asserted, it is reasonable and appropriate to demonstrate, to the extent that it can be done, all of the forensically demonstrable properties. {Quantified, Time framed, Tangible, Unmitigatable, Uninvited, and Causal} However; the necessary

evidence to demonstrate these may not be easy to come by, and those challenging the case may try to prevent this from being done by various legal maneuvers, to the extent that they are aware of the issues and can limit their exploitation. At the same time, the job of those challenging damages is to challenge each of these properties in the hope that one or another of them will be affirmatively demonstrated to be untrue or that other side will be unable to demonstrate them to be true to the required standard of proof. In some cases, and in some jurisdictions, strategy and tactics of the case may dominate these concerns, and some lawyers, clients, or examiners may not want to or not know to apply all of these issues. In special cases, some of these may not apply, and of course everything in the law is subject to change with time and judicial decisions. Nonetheless, these demonstrable properties are generally at issue and the thorough participant in the process will seek to explore and understand them all within the limits of the schedule.

Specific traces of import to attribution of damages tend to be audit data and other normal business records combined with numerical analysis in context. The normal business records assure that the traces and events can be shown to be reasonably reliable and trustworthy and free from alteration or corruption. Someone has to testify as to how these records came to exist, were stored, processed, transported, and so forth in order to authenticate them. Since the records of interest to calculations are being offered for the truth of what they portray, they must be brought in through this path in most cases. Some records created on a custom basis for a particular matter are problematic in terms of being admitted, and their admission will have to be done by having an expert testify as to their properties and justify their use subject to challenges by opposing counsel and their experts.

Quantification of damages

As a general notion, the problem of quantification depends on the measurements available. The measurements available are typically derived from the traces and events. There are two classes of cases that may be distinguished; (1) continuous damages with activity and (2) step function damages with activity.

- Continuous damages can presumably be shown to relate to quantities of activities shown by traces, so damages may be reasonably prorated to relative quantities of activity shown by the traces using a court approved calculation method. Linear damages with activity levels are typically easiest to establish in the continuous case. For example, if computer time or bandwidth is consumed so that less is available to other uses, but the uses continue at reduced performance, use is deprived proportionately with performance reduction. Traces indicative of normal and reduced performance, such as packet logs, access logs, or other similar traces, may be used to prorate damages.
- Step function damages have a qualitative relationship between the presence of traces and events and the value of the damages. In this case, the presence of adequate traces to show harm may be adequate to establish a defined quantity of damages regardless of the quantity of those traces or the activities they demonstrate. For example, if it can be shown that traces and events causally relate acts of a party to total system outages over a period of time, the full value of the use of the affected systems over that period may be assessed.

The continuous damages case

If memory usage is charged per byte per hour and inadequate memory is available for a process, then the user may not use the system for that purpose and the lost revenue can be measured as the lost revenue, assuming it is not otherwise recovered. If a resource is normally fully consumed and cannot be fully consumed for a period of time, then the loss is, presumably, proportional to the usage differential for the period of time. This of course assumes that what is lost is never regained, but this is rarely true and difficult to show. The more common case today relates to service level agreements (SLAs) with fulfillment related charges and failure penalties.

Defined service levels may or may not exist. To the extent that they affect a business, they can and should be measurable if they are meaningful to the business. If the business has no way to measure

them, it is difficult to assert that there are damages associated with their reduction at a detailed level. If services are not affected to the point where they cause failure to meet defined service levels; regardless of the actual incremental effects of additional disk usage, processor, memory, network loads, or other resources; they cannot be reasonably asserted to have a damaging business effect. It is, presumably, the responsibility of the party claiming damages to assert some basis for those damages in some measurable metric. The issues are then, presumably, presentable as consisting of:

- **Resource:** The resources that can reasonably be claimed as being affected under an SLA are those resources reasonably required in order to provide the services identified in the SLA. Typically, these are the obvious resources, but in order for them to be used, they typically depend on the less obvious ones. Disk usage and information nodes (inodes: some file systems limit the number of files that can be present) are examples of obvious and interlinked less obvious resources. However; modern SLAs rarely include anything inobvious. Available disk space might be claimed as the deprived resource, is sometimes within SLAs, and the traces to support such a claim might include inodes used, files used, and other related resources that are recorded in traces in one fashion or another.
- **Damage Type: Deprivation** Services unavailable to meet defined service levels because of acts of the party claimed to have caused that unavailability seem likely causes of measurable damages. This implies SLAs and contract terms with sanctions or some other basis for establishing the deprivation. In addition, to the extent that such deprivation was not reflected in customer complaints, sanctions, or loss of business, it is problematic to measure damages.
- **Economic cost: amount claimed** This is a numerical value that is the result of some calculation that can be justified by the basis. The amount claimed under deprivation cannot reach the total cost of the resource. This would be a claim of conversion. This may be rationally calculated by lost revenues and sanctions paid attributable to the claimed harm as shown by the traces and events.

- **Basis for claim:** For an economic loss to be claimed, some revenues that would otherwise have been gained would have to be shown to have been lost, or some cost that otherwise would not have been incurred must be shown to have been caused. For deprivation, the SLA would normally be the starting point for analysis.
- **Quantified:** The loss is typically quantified by showing that some set of customers complained and were compensated or subsequently dropped the service(s). Evidence of prior payments may be used to show economic impact and statements of prior customers may be used as evidence as to cause. If increased costs were incurred, these can be listed and attributed to the extent that they can be shown to be caused by the asserted actions and attributed to that party. For example, computer-based measurements in the form of analysis and interpretation of audit records, server logs, and traces of content or usage patterns might be used as evidence of a customer being inundated. But to show that general performance was affected, calculations relating traces to impacted performance to the point where service levels could no longer be met may need to be shown. The calculations would likely be based on the SLA and use the calculation method specified in the SLA and the traces and events required by the SLA to make those calculations. If this sort of detail is not available, the contract defining the SLA is problematic in that it does not define a method for proof of meeting or failing to meet the SLA. A typical measurement method might be the time between a request and its servicing, the number of bits per second of bandwidth available to the client, or the amount of disk storage available for use by the client. The traces and events used for measurement might be results of router flow records, time traces produced from server logs, disk utilization reports generated by accounting records, and so forth.

Putting time frames on damages

Computer records are typically very useful in establishing time frames. For example, audit records and records from servers, browsers, clients, file systems, and traces from content of files,

messages, and documents, typically include time and date stamps. To the extent that damages can be shown, the records that form the basis for those damages typically include time stamps that allow the times of damages to be shown from those records, subject to the accuracy of the time-related traces. The major constraints on timekeeping stem from inaccuracies in system clocks, time zones and differences, format differences, and recording anomalies. The processes discussed in the chapters on Analysis and Interpretation will typically be used in this process.

Given that times are properly associated, performance indicators used to show damages, such as increased times for performing standard processes, increased network packet losses, increased processor workload, increased paging, errors indicative of resource exhaustion, and so forth, may all be linked to the times of arrivals, processing, storage, and delivery of things reflected in traces. For example, when thousands of emails arrive and are processed within a few minutes and the number of file handles is simultaneously exhausted, this provides a time frame in which it might be asserted that the presence of all these emails caused the exhaustion of file handle resources within the operating system, leading to the failure of a process that provides some customer service that is normally provided. This then provides a potential causal link to the damage type of deprivation and the time frame over which the deprivation caused actual damages.

Tangibility of damages

Tangibility differentiates between (1) intangible sorts of harm; like the feeling of discomfort when someone sends an undesired message to a server, electromagnetic emanations associated with a user's increased numbers of processes, or the changes in a stored value from 0 to 1 or 1 to 0 in otherwise unused areas of memory; and (2) tangible harm, like lost revenues, inability to provide contracted services resulting in increased costs in the form of increased complaints, payments made for failure to meet SLAs during the times associated with the harm, or fees paid to providers for adding bandwidth during an attack.

For example, if a deprivation is asserted from consumed bandwidth during a part of a day when there were no other activities underway

and no usage charges are accrued, then the deprivation is typically intangible at best. If a computer crash can be shown to be caused by use, then the loss of services and costs of repair are more likely to be considered tangible and demonstrable as damages.

Tangible damages typically leave traces in forms like logs of regular processes that are delayed, error messages relating to resource exhaustion, time stamps on files indicating unusual wait times for certain processes, and telephone and help desk logs of customer complaints. To the extent that such records are not kept or generated, it is harder to show tangible damages. To the extent that these mechanisms are disabled or logs discarded or not produced, this is also problematic, in that traces not available cannot be used to confirm or refute hypotheses or events related to tangibility of damages.

Showing mitigation of harm

To the extent that harm can be mitigated by reasonable and prudent practices or timely action, there is a responsibility of a party asserting a claim of damages to mitigate that harm and the potential to charge the cost of mitigation as damages to the extent that it is not just a matter of reasonable and prudent practices. For example, if the traffic causing problems in a server need not be accepted and can be readily blocked by some technical approach, then the failure to use this technical approach when it is known to be feasible and not excessively costly, is a demonstrable failure to mitigate harm. To the extent that the harm could have been mitigated by these actions, damages are not likely to be granted.

Failure to take reasonable and prudent measures to protect might also be considered negligence or gross negligence, and to the extent that this is relevant to the issues or affects innocent 3rd parties, mitigation of harm should be undertaken to prevent 3rd party damages. Many DFE examiners are very knowledgeable about issues in information protection, but the examiner who isn't a true expert in this area with the experience, knowledge, training, education, and expertise required to back this up, should refrain from opining on reasonable and prudent practices.

The DFE examiner may analyze and interpret traces and events so as to identify what practices are in place. For example, traces

commonly show consistency with changes made to control mechanisms, installation and operation of protective mechanisms, software updates, and similar activities, and may include trace information that goes to the timeliness of these actions relative to the time frames identified for damages. By correlating these traces to events, interpretations relating to what was done to mitigate harm may be undertaken.

Attempts at mitigation generally come as combinations of policies, procedures, and technical countermeasures, and are generally introduced as events. For example, if there is a policy against signing up to certain types of mailing lists and employees are properly trained to request removal from such lists, and that policy is enforced, then this provides reasonably strong evidence that there is an attempt to mitigate harm associated with signing up to those lists, even if it might not be considered adequate in terms of mitigation of damages from such lists. If technical measures such as the use of an external provider or the purchase and proper use of a blocking device to mitigate undesired traffic are in place, this is likely to produce traces demonstrating attempts at mitigation. On the other hand, if such countermeasures are readily available and unused or intentionally configured so as to not operate, this is more likely to be viewed as an invitation.

Demonstrating that the actions are uninvited

In some cases, parties to legal actions have intentionally not mitigated harm and actively configured their systems to accept things that normally would not be allowed. For example, email providers have configured servers to accept emails to users that don't exist, have created fictitious accounts for the sole purpose of accepting unsolicited commercial emails, have taken over accounts of former users to collect unsolicited commercial emails, and have formed agreements with 3rd parties to send unsolicited commercial emails to them. This then goes beyond the realm of failure to mitigate harm into the realm of intentionally inviting the messages knowing or reasonably expecting that they will cause harm that will then be used to pursue legal action. This is in the extreme end of inviting the damages, is likely to result in a ruling that the messages were invited, and may even result in sanctions.

Traces are often available of this sort of activity in the form of the asserted messages and their recipients which can be compared to traces or events indicating the authorized users of the systems, the paths through which messages were sent, configurations of servers which indicate whether they were intentionally modified to allow what would normally be blocked, log files, metadata, and other similar traces that indicate the nature and type of errors ignored, detected, and acted upon. A similar situation might exist in the attempt to assert damages from the activities associated with the use of a honeypot or similar deception system.[326,327] Such systems are designed for the purpose of receiving malicious activities. As such, any attempts to break into them and successful activities along those lines can hardly be claimed to be uninvited.

A more common situation among legitimate email providers is that their users inadvertently invite messages, for example, by signing up to a service or connecting to a Web page. In this sort of situation, a provider suing a sender would have to demonstrate that these messages were not invited, presumably by having their customers so indicate, and then these customers may become subject to cross examination if these claims are doubted.

Enterprises often enter into contracts that include rights to send commercial messages, and many products in common use include contracts with language granting the other party rights to communicate. In such cases, they literally invite those communications unless they put limits in the contracts and provide for penalties associated with violations. In these cases, the damages will be defined by contract. They only have to be measured properly.

Large volumes of usenet messages are commonly downloaded from servers when users sign up to services, and this often causes performance problems, but the usenet provider cannot be held liable for damages related to these cases because the user invites these messages. As a technical matter, many providers then limit usenet downloads to their own servers and cache usenet

326 The Deception Toolkit is an example of a mechanism that may be considered an invitation. See: http://all.net/dtk/index.html

327 The Honeynet Project is an example of a class of systems that is designed to be an invitation. See http://www.honeynet.org/

messages to reduce external bandwidth consumption in this area. This is an example of preventing invitation and mitigating harm, and it can be extended to other areas, but only to a limited extent. Many companies also limit messaging services like Internet Messenger so that it cannot exceed a given bandwidth and so that it will not allow files to be downloaded.

Another approach is contractual. Companies can contact other companies and request that they not send commercial emails, or can act on behalf of their users, if the users allow this by contract, to automatically try to remove them from email and other lists. There are commercial services that do this, and of course for something like $10 per month per user, a 3rd party service provider will detect and remove most unsolicited commercial emails. But again, if the company using such a service tells the service provider to allow certain classes of email to pass, such as emails to otherwise unassigned email addresses, and then accepts them even though they have no legitimate users with these addresses, then they are inviting the emails in and will have serious problems showing damages in court against competent counsel.

Demonstrating causality

Demonstrating causality is based on a combination of internal and external attribution of causes to effects. In a legal setting, causality only has to be shown to a standard of proof associated with the legal issue at hand. For a criminal case, the standard is typically "beyond a reasonable doubt", far higher than for a civil matter, where only the "preponderance of evidence" is typically required. External attribution is the subject of much of the rest of this chapter, and in this discussion we will assume that traces and events provide the means to attribute actions to individuals or organizations to the required standard of proof.

For internal attribution in a civil matter, it might be adequate to show close correlation, such as to demonstrate that at the identified time frames given by traces from asserted messages, audit records are consistent with user response time being slowed significantly and complaints, as recorded in the help desk call logs, increased in the same time frames starting after the cause and ending after mitigation. For a criminal case, it might be necessary to show more,

such as to demonstrate that the specific activities actually result in the identified consequences.

Correlation is not causality, even though it may often be adequate for a showing. However, a negative correlation in which the asserted deprivation is shown to be present in larger amount when less of the cause is present, or a "cause" time after the "effect" time, is almost certain to refute any claim of causality as to damages. This should be readily shown with the same records that are used for attempting to demonstrate causality in the first place. Failure to produce the relevant traces may prevent the opposing side from making such a showing, and this is problematic. Demanding such traces and not getting them leads to issues related to document retention and disposition, and failure to properly retain records when there is a reasonable expectation of a legal action has led to sanctions and lost cases.[328] Inability to show time ordering is certainly problematic as to causality, and failure to provide records that could allow other parties to show lack of causality is also problematic. Judicial rulings on the failure to provide evidence that should have been preserved and that could prove a lack of causality are the only real way to force this issue.

As a different example, if a computer crash can be shown to be caused by an activity, then the loss of services and costs of repair are more likely to be tangible and demonstrable. But if the crash destroys relevant traces, it may be harder to prove causality. For this reason, it is better to have audit records sent to independent servers for collection and storage. This provides better separation of duties and a better chain of custody for evidence in most cases, assuming the process can be shown to be reliable and proper records are kept for evidentiary purposes.

A diligent effort to secure evidence

A diligent effort to secure evidence is not something that can be defined in advance for all cases, but to get a sense of what might be considered diligent for attribution of damages to causes, the following is put forth as a starting point:

328 The Sedona Guidelines: Best Practice Guidelines & Commentary for Managing Information & Records in the Electronic Age, A Project of The Sedona Conference Working Group on Best Practices for Electronic Document Retention & Production, September 2004 Public Comment Draft.

- **Time framed:** The time frames of effects are demonstrable through audit trails and other related traces. For example, the same traces used to show that service levels were reduced by indicating time to perform SLA-agreed tasks may demonstrate that more time is taken in performing other tasks, changes in volume of processing performed, delays in completion of tasks, or other effects specifically called out in the SLA. The items in the SLA are events that analysis and interpretation links to the same traces called out in the SLA as proof of fulfillment or to show non-fulfillment. Times associated with SLA failures should be reflected in these records, as if they are not, SLA enforcement becomes problematic. Router flow reports, server logs, and disk utilization reports are typically generated with time and date information, and this provides additional traces of the timeframes for damages.
- **Tangible:** To the extent that there are records kept and that an SLA defines services, damages may be tangible as described in the agreement. Unless there is some specific item that it not tangible as defined by law, the items specified in the SLA, measured for the purposes of the SLA, and used as a basis for charges, should be adequate to demonstrate that damages are associated with tangible items. However; to the extent that the SLA defines tangibles that do not in fact relate to the useful functioning of the service, they are unlikely to be accepted as damages, even if a penalty was paid. Just because an SLA indicates that all unused areas of disk must hold "zero" bit values until used through the SLA does not mean that a 3rd party is liable for damages when their actions cause such disk areas to be altered.
- **Unmitigatable:** Mitigation of damages is an area where proof is relatively easy to demonstrate, given that diligence is used in the management of systems. The sequence of events recorded in contemporaneous records, and traces of those events, such as email exchanges relating to the identified activities, log entries in operator logs, and other similar material, is readily available in most enterprises. Evidence of reasonable computer security programs are

typically used in these cases, and include policy documents, control standards, procedures, documentation associated with the security program, and so forth.[329] In the case of messaging, the configuration settings of servers, firewalls, and other related mechanisms, regular updates to keep software current, payment of fees to external service providers to control UCE, and similar records, show diligence in general. Specific efforts related to the specific damages associated with an incident are typically shown in written reports such as post-incident reporting, records of efforts to mitigate harm, exchanges with third parties relating to the harm and how to mitigate it, and so forth. The time history of the incident is also typically produced for significant incidents to aide in learning how to better handle future incidents. An incident response team is typically used in companies, and the records of the team, the help desk records, and records associated with work flow processes are all available to help show that reasonable and prudent efforts were undertaken to mitigate harm. These are also potentially evidence that can be used to show the cost of mitigation and charge this as damages.

- **Uninvited:** To demonstrate that actions were not invited, the same information that is used to show that the damages were mitigated typically support the technical aspects of invitation. Servers that are explicitly configured to allow unusual activities are, potentially, invitations for those activities. In addition, these sorts of activities can usually be readily stopped at any time, and failure to stop these activities when they are perceived to be causing harm implies a lack of diligence in mitigation. Invitation from a human standpoint implies taking testimony about human actions and is beyond the scope of this book.
- **Causality:** Causality flows from the traces and events just given. However, the relationship between cause and effect are typically only contributory and not exclusive. For example, if a computer system has only a fixed total amount

329F. Cohen, Enterprise Information Protection", ASP Press, 2008, ISBN# 1-878109-43-X.

of disk space and undesired messages consume only a portion of that space, when space runs low and damages are incurred for a period of time, there may be many sources of undesired messages involved. If one source is identified and that source's messages consumed 20% of the disk usage associated with the undesired messages, they are only part of the cause of any damages, and presumably are only responsible on a prorated basis. Causality also requires time ordering in that cause must precede effect.

Most trespass damages are low valued

Because of the relatively low cost of memory, storage, bandwidth, and other computer-related equipment, damages from loss of utility may be viewed as very low. Again, ignoring indirect consequences, actual damages cannot be as high as the total value of the assets interfered with. Given that typical computer-related devices have life cycles of 2-3 years, a complete outage of an entire system for a month cannot produce damages from deprivation in excess of the cost of a replacement system divided by the 24-36 month normal life cycle. If a system with 1 Terabyte of disk, 16 Gigabytes of RAM, a fast processor, and a 10 Gigabit Ethernet interface costs $2400, the direct loss from deprivation for a month cannot exceed $100. If a disk has to be purchased to extend available storage, the damages are likely to be very small because the previous disk, presumably, had already lost a portion of its useful life and might be repurposed for other use. A 1Tbyte disk costs about $100, and this corresponds to only about $4/mo if it is entirely consumed by the attributed acts.

Table 7.6 shows an example of damages that might be associated with network availability reduction caused by bandwidth consumed by undesired traffic. In this theoretical example, SLAs were broken with multiple users who were rebated funds based on the SLA and the lack of fulfillment caused by the traffic. Bandwidth is charged on a pro-rata basis and the added bandwidth resulted in added charges.

Resource	Damage type	Amount	Basis in forensic properties
Network bandwidth (100Mb/d of lost use).	Deprivation customers unable to get SLA bandwidth	$150	Consumed bandwidth in specified blocks of time with normal users unable to get SLA guaranteed services during those periods. 30 users were rebated $5 for inadequate service
Network bandwidth usage	Added costs of added use	$100	Added costs of bandwidth actually paid based on per byte add-ons to monthly bill
Operating costs	Added costs of mailing out rebates	$30	Envelopes, checks, and mailing costs came to $1 per rebate sent out
Support costs	Added costs of servicing customer calls and rebates	$100	30 support calls at 10 minutes each handled by an external provider and charged back at $20/hr

Table 7.6 - An example partial claim for actual damages

Note that the added costs are all direct results of the deprivation claim and are asserted in addition to the direct damages associated with the trespass tort. These are compensatory damages identified as part of making the injured party whole.

Obviously, for $380 it would be a waste of time to try to sue. However, if statutory damages for the messages came to $1,000 per message and no statutory damages are allowed unless actual damages are shown, then a $100 million law suit would likely justify the effort to prove the $380 in actual damages that enables the suit to proceed.

Another area where nearly continuously variable damages might be identified is in power usage. While some power usage differs from instruction to instruction, and this can be measured if desired,[330,331]

330 F. Cohen, "A Matter of Power", Managing Network Security, Elsevier, 2001, located at http://all.net/journal/netsec/2001-07.html

331 Vivek Tiwari, Sharad Malik, Andrew Wolfe, and Mike Tien-Chien Lee "Instruction Level Power Analysis and Optimization of Software", 9th

the differential power between instructions executed is truly insignificant in terms of actual costs to the business. To get a sense of the extent to which these damages are minimal, measurements were taken in or about 2000 to examine the effect of power consumption and performance in commercial off-the-shelf computers at that time using a power meter. Other than sleep modes, to 3 digits of accuracy, no difference was detected in power consumption between computers idling and computers performing computation. The power consumption differential of 16% at the level of individual programs is far less than the differentials shown in processor selection, so as a relative cost, a party who has not taken the time and effort to make determinations related to processor selection cannot reasonably assert that they are diligent or care about this differential power cost. Substantial power savings can be achieved by processor selection, power supply selection, memory size, disk selection, and by far the largest contributor, video display technology. The costs associated with power differences in computer selection are on the order of $50-$150 over the lifetime of the computer for $5-$10 differences in acquisition costs. Lowering power consumption also saves on air conditioning.

Other effects, such as additional air conditioning usage, wear and tear on disk drives, and so forth are also measurable, even if they are relatively small and may not be worth the effort in most cases.

The big problem in terms of applying the tort of trespass to these sorts of costs is that they cannot be readily shown in most cases to have caused physical damage, they are not cases of conversion, deprivation is only an issue if this truly small amount of added use caused a power outage or other similar event, and that would be difficult to show causality for. The lost value, if any, would also be trivial, and lost rights are not readily apparent.

All of this adds up to a simple conclusion. Unless and until some threshold is approached or exceeded that results in an actual added cost or an actual failure to meet a defined level of performance relative to an SLA, direct damages for activities that consume time or space on a typical server can only be considered minimal.

International Conference on VLSI Design - January 1996.

Attributing damages at a step

A step change in damages occurs when an operator is required to make a purchase or is unable to fulfill a defined service level and that change can be attributed to an external cause. This, it seems, is the major, if not the only case, in which damages such as the requirement to process excessive usage or traffic requests can reasonably be attributed to a cause. Given that such an event in fact occurs, it is incumbent on the party claiming damages to identify the damages in a meaningful way. Clearly, there are direct costs, such as the cost of purchasing additional disk space, bandwidth, memory, or the lost business directly resulting from the inability to move information. But it may be less clear that these damages are a result of the undesired use as opposed to other factors. In many cases, disks contain large quantities of deleted or otherwise unused content that accumulates over time and is not removed. Many users make backups on the same disk they are using and compile large volumes of files and other content that may merely require a cleanup. And in many cases users want more disk space, processing power, bandwidth, or other resources for other purposes.

Many businesses do regular upgrades of hardware, software, and other infrastructure elements and these replacements may not be directly related to external causes. In other cases an otherwise necessary upgrade may be sped up by the presence of large volumes of external content, but is not directly caused by it. In any and all of these and similar cases, it is incumbent on the parties to battle out the causes of change-overs, the regular schedules of upgrades, and so forth, to generate a settlement as to what differential damages may be.

At a step, a formula will somehow have to be developed and accepted by the courts to attribute damages to parties involved. Such a formula would reasonably be expected to take into account prorated attribution of damages to all involved parties and also take into account the portion of the lifecycle of all components involved. Again, the financial impact of the tort with respect to trespass will likely be very small for most situations encountered today.

The nature of control

Attribution implies, to some extent, control over actions leading to consequences. That is, when attributing actions to actors, there is a distinction to be made between an actor who:

- (1) through intentional actions over which they have control, knowingly and intentionally causes an event sequence to take place, or knows or should know that an outcome will take place,
- (2) through intentional actions over which they have control, unknowingly or unintentionally causes an event sequence to take place, or
- (3) through accident or events not within their reasonably anticipated control, unknowingly and unintentionally causes an event sequence to take place.

This is the concept of the difference between "mens rae", the guilty mind, and happenstance. In the technical world of digital systems, there are some specific issues that may shed light on these issues.

Instructions versus intent

Programmable digital systems are automated mechanisms that, when properly operating, execute the instructions given to them. While people may, at times, anthropomorphize computers to associate seemingly human properties to them, computers are not people, and cannot today be reasonably said to have intent in the sense that people do.

It is common parlance to use the word "command" when describing interactions with computers. The person types a command and the computer carries it out. Thus, it could be argued that any interaction between a computer and a human involves a human with intent issuing commands and a computer receiving and executing those commands. When people make mistakes in entering commands, or when the computer program interpreting the inputs does not properly interpret the intent expressed or intended by the human, the computer will nonetheless, act on the input it was given, without regard to intent.

Digital systems interpret whatever inputs they get, and perform whatever functions they are programmed to perform, based on the manner in which they are designed, implemented, configured, and operating.

- For **general purpose mechanisms**, this means that the input **can express any intent** within the capability of any computer, **and** if properly expressed, **the mechanism will carry out that intent**.
- For **special purpose mechanisms**, no matter what the intent of the person ultimately responsible for the input, in normal operation, **the mechanism can only carry out the intent of the designers** as expressed by the implementation and configuration.

A special purpose mechanism can only carry out the intent expressed in the input to the extent that it is within the intended purpose of the designers and the constraints of the implementation and configuration. Thus a difference between general purpose and special purpose mechanisms is that:

- **General purpose mechanisms** are designed to **allow any intent** expressed in the input **to be carried out**.
- **Special purpose mechanisms** are designed to **allow only the intent of the designer to be carried out**.

What it means to be in control

A sense in which the term "control" may be reasonably seen is in the notion that control requires that intent can be expressed and that expressed intent is carried out. That is to say:

- If a party cannot meaningfully express intent with regard to any specific act, they cannot reasonably be said to have been in control of that act.
- If a party's expression of intent with regard to any specific act is not or can not be acted upon, then they cannot reasonably be said to have been in control of that act.
- If a party can express intent with regard to a specific act, and if their expression of intent regard to a specific act is or can be carried out, then they may be in control of that act.

As summary, acts not expressible by the interface or not within the manifold of the expressed intent of the designer, cannot be reasonably said to be controlled by the user of that interface or mechanism. For example, and without limit:

- The owner of a computer who can configure, program, and operate it so as to perform acts of their choosing, may be in control over that computer and thus potentially attributed as the cause of its actions.
- A user of a Web server who uses the server in the normal way it operates and within bounds of normal and reasonable usage, cannot reasonably be asserted to be responsible for the space taken up by that server's logs of that user's use.
- An actor who exploits a vulnerability in a Web server to cause that server to act outside of the manifold of the expressed intent of the designer and operator, may indeed have taken control over that mechanism and thus be attributed as the cause for resulting actions.

In the first case, control was in the hands of the owner, and thus their acts could potentially be considered causal with respect to what the computer did with regard to the things they controlled.

In the second case, the user of a Web server using it in the normal manner, has no control over whether or to what extent that server logs or does not log their activities, because (1) there is no syntax by which such a user can normally express an intent to have their actions logged or not logged, and (2) even if there was a way to express such a thing, the normal logging of Web servers does not respond to user requests so as to enable or disable logging, unless they are somehow specially privileged to do so, and using an administrative interface.

In the third case, the normally special purpose interface to the Web server is bypassed by the actor, thus changing it from a special purpose interface and/or mechanism to a general purpose interface and/or mechanism. At that point, the actor gains control over the server, or some part of it, that is outside of the expressed intent of the designer and operator.

Overall attribution

There is a cumulative effect of a multitude of attributions of traces to individuals, programs, platforms, events, and devices, that can accrue when multiple methods are used in concert. While each may, on its own, have limited probative value, depending on the jurisdictional restrictions, the standard of proof, and the qualities of the different attribution methods, results may be made available to the triers of fact and weighted as they wish.

Redundant records as indicators

One way to increase the probative value and certainty of attribution is to provide an accumulation of relevant traces, events, and interpretations that consistently support the attribution, and at the same time systematically eliminate alternative events and hypotheses, by showing that they are inconsistent with asserted events and traces. To the extent that this can be done exhaustively relative to some model of causality, a level of certainty with respect to that model is, perhaps, attainable. While a single refutation may counter all of the attribution efforts involved, redundant attribution paths may survive multiple refutations that eliminate one or more of the hypotheses or sets of traces and events, without refuting the overall attribution, or all paths from cause to effect.

Mens Rae and attribution

The state of mind of the actor may also be something that can be supported through the attribution process, particularly when there is hidden content involved and it can be tied to an individual. While it might be accidental or a moment of weakness when an individual performs an act that is subject to a legal action, if there is a consistent timeline of traces and events available that shows actions taken to hide the act, this goes to the guilty mind and intent of the actor.

For example; if the actor is attributed at some level of certainty as having (1) loaded software capable of performing a hiding function, (2) subsequently performed identified violating acts, (3) then hid the results of those violating acts, and (4) then deleted the obvious records of those violating acts; it will be difficult to argue that the actor did not know that the acts were violating, did not plan on undertaking those violations, or did not know what they did.

The sequencing of acts that is consistent with traces and events is quite powerful, and even if the strength of the attribution is less than ideal, the overall pattern of behavior shown by the events and traces is more likely to result in a successful claim of state of mind.

Defending against such claims can be problematic unless the attribution can be systematically defeated step by step or can be defeated because of inadequate redundancy in its formation. For example, suppose all of these attributions are based on the hypothesis that a party sent some sequences of messages, and that attribution is based only on the content of those messages, including the URL on a server owned by the party under scrutiny. Clearly, this may be a false attribution, since anybody could include such a URL in a message. This doesn't tie an individual to the acts.

Verifying the integrity of attribution mechanisms

The mechanisms relied upon for attribution are also potentially problematic in that their realizations and the methodologies they are based upon may be faulty or inadequate to the weight given them by the examiner.

One challenge is that systems and mechanisms change with time. As a result, the particular instances of mechanisms change over time, and the characteristics of those mechanisms today may differ from what they were at the time of interest to the legal matter. Contemporaneous traces must be used, and assertions about mechanisms and methods should be reconciled with their state at the time of interest. Break-ins to systems in the intervening time may have corrupted the traces or the mechanisms, and if the mechanisms weren't working properly, the results cannot be relied upon to be accurate. Mechanisms may also have been repaired since they were corrupted, so that between a "clean" installation and the time of trace collection, the mechanism could have been corrupted, and between mechanism corruption and trace collection, the mechanism may have been repaired. Thus traces of the faulty mechanism or resulting trace corruption may not be found.

Even if the mechanism is unaltered, the way it worked in situ may differ from the way it is believed to have worked. For example, network-based timing information may have been less accurate at a prior time because of traffic loads, equipment differences, or usage

pattern differences. If timing is important to the attribution mechanism and cannot be established properly, it is potentially problematic. The same sort of thing may be true for a process within a computer, a log file, or any other mechanism involved.

Attribution has the potential of abuse, in that malicious actors may intentionally create false attribution information and seek to cause the mechanism to falsely attribute actions to innocent third parties. This is the case when someone uses another individual's account, either when they are away from their desk or by using their authentication device or password. Covert channels and Trojan horses may allow a malicious actor to surreptitiously use another individual's authenticated session to act so as to attribute acts to the innocent third party. There are many other mechanisms that can cause attribution to fail to properly attribute actions to actors or effects to causes. For this reason, in making assertions about attribution, the examiner should be careful about statements made.

While redundancy may be very effective in overcoming these sorts of attribution limitations, if the redundancy is not separate and different, then common mode failures may exist that cause the redundant mechanisms to fail to achieve independence required for improved surety. Assertions about independence of mechanisms should take common mode failures and other interdependencies into account.

Statistical and other combinations of attributions are problematic and should be avoided unless the examiner is sufficiently knowledgeable about all of the facets of statistics and the mechanisms required to make proper assertions and back them up with a scientific basis. Otherwise, these sorts of statistical claims are asking for a challenge. Prejudicial effects also ultimately get weighed against probative value prior to admission of attribution information. To the extent that meaningful metrics may be applied to show the scientific value and accuracy of results, this goes to the probative claim.

Verifying that the attribution goes in the right direction

In order to have causality, a chain ***from*** the cause ***to*** the effect is required. A chain that goes part of the way from the cause to the effect and part of the way from the effect to the cause, does not

definitively establish causality, regardless of how compelling it may seem to be. Similarly, a broken chain with links in the middle but an incomplete path does not definitively establish causality. As obvious as this may seem, a chain that does not go in the proper direction and every step of the way may seem to be quite compelling if not thoroughly and thoughtfully examined.

As an example, suppose the examiner has traces that are consistent with events, and that the events assert that:

- Party A sent a set of items,
- Party B received all of those items,
- There is a contiguous path showing the items went from party to party starting at party A and arriving at party B, and
- Party B provides a signed statement: "Party A sent the items to party B".

Do you conclude that "Party A sent the items to party B"?

STOP AND THINK BEFORE CONTINUING

If you conclude from this set of facts that party A sent the items to party B, you are making a mistaken attribution. Why is this?

STOP AND EXPLAIN WHY BEFORE CONTINUING

The problem apparently stems from the substitution of consistency for truth. Just because the statement of party B is consistent with the other assertions, does not make it true. In fact, the causal chain is not complete. To show this, let's add some new information. Suppose additional events come to light in that Party A asserts:

- They did not send those items to party B, and
- Except for the statement by party B that "Party A sent the items to party B", everything else is true.

This too is apparently consistent, except in that party B's statement is inconsistent with party A's statement. Given that there is no particular reason to trust party A more or less than party B, the attribution should be given no weight. Is this right?

STOP AND THINK BEFORE CONTINUING

If this still isn't clear, suppose that party A also states that:

- The items were actually sent to party C, but party B intercepted them

or perhaps party A might state that:

- The items were actually sent to party C and party C sent them to party B

There may be many other statements that could clarify things.

This example should provide a fundamental challenge to attribution in that, unless all alternatives have been exhausted, there may be a reasonable explanation for all of the traces that is consistent with some or all of the events, and that refutes the attribution.

Checking overall results against information physics

As a rule of thumb, the examiner should try to check overall attribution results against information physics to make certain that none of the laws of that physics are apparently violated in the overall attribution. While checking each step for information physics is, in some sense, inherent in applying the methods described, the combination of steps that individually meet all of the known information physics requirements may not, in the aggregate, meet those same requirements.

An analytical process for showing causal chains in attribution

This analytical process provides a systematic means for an examiner to determine whether or not an actor appears to have had adequate control over a chain of events to be said to have knowingly and intentionally committed a prohibited act. This analysis largely ignores details of the attribution and trace consistency issues, in favor of linking to results of examination. The analytical process is as follows.[332]

For each identified potential causal path $\mathbb{P}$ from mechanisms $m \epsilon M$ in the control of suspect to a violation:

- Identify $\mathbb{P}=(m_1, ..., m_n) \epsilon M$, (a sequence of cause ($\mathbb{C}$) → effect ($\mathbb{E}$) mechanisms that constitute the path $\mathbb{P}$).

332 F. Cohen, "A Method for Forensic Analysis of Control", IFIP TC-11 Computers and Security, V29, #8, (2010) pp 891-902.

- ∀m∈ℙ, determine whether m is general purpose.∈
  - If m is special purpose, examine the syntax and semantics of m to identify direct or indirect means to affect asserted effects. If no such means exists, rule out ℙ, otherwise identify and document (ℂ→𝔼)∈m.
  - If m is general purpose, examine the envelope of m (recursively), identify direct or indirect means to affect asserted effects. If no means exists, rule out ℙ, otherwise identify and document (ℂ→𝔼)∈m.
- ∀ remaining ℙ, ∀m∈ℙ, identify traces (t∈T) probative with respect to (ℂ→𝔼)∈m and search for such traces. If ∃t∈T that are inconsistent with (ℂ→𝔼)∈m, rule out ℙ. Otherwise, if ∃t∈T consistent with (ℂ→𝔼)∈m, confirm m. Otherwise, indicate that traces do not confirm or refute m.
- ∀ remaining ℙ, ∀m∈ℙ, identify traces probative with respect to exploitation or bypass of m and (ℂ→𝔼)∈m, and search for such traces. If ∃t∈T that are consistent with exploitation or bypass leading to (ℂ→𝔼)∈m, identify possible alternative explanations of m. Otherwise, indicate that traces do not confirm identified alternative explanations of m.
- ∀ remaining ℙ, ∀m∈ℙ, attribute acts to Party.

Identifying and exhausting the forensic procedures (P) is infeasible in most cases, so examiners identify feasible and available p∈P for each of these steps by using their education, training, experience, skills, and expertise.

The case for the accuser

For the accuser, to the extent that this process was undertaken, full details should be provided of each step in the process so as to adequately support the claims. To the extent that there are redundant paths by which the claims may be shown, this is potentially problematic in that, presumably, only one thing actually occurred. Several conditions arise other than a perfect and fully completed analytical process. As we saw earlier, the total set of procedures that can realistically be performed is far less than the total number of possible procedures that could be performed for

any nontrivial case. Traces may not be available $\forall m \epsilon \mathbb{P}$, and thus many remaining paths may exist, any of which may be feasible. Thus the case generally consists of statements of the form:

> "It appears that [$\mathbb{C} \rightarrow \mathbb{E}$, ..., $\mathbb{C} \rightarrow \mathbb{E}$]."

with the basis for this appearance provided at the level of detail available.

The granularity of $\mathbb{P}$ is also potentially an issue. While $\mathbb{P}$ could potentially be explored at the level of each hardware component, it is normally examined at a far higher level. Typically, $\mathbb{P}$ is explored at the level of actions by the suspect with technical details supporting the claims of these actions at whatever level is required in order to identify relevant traces or events. Again, there are almost certainly details not explored, and the $\mathbb{C} \rightarrow \mathbb{E}$ sequence forming $\mathbb{P}$ is incomplete, even if it is reasonably convincing.

The case for the accused

Given $\mathbb{P}$, the examiner for the accused is responsible for identifying the limitations of, and flaws in, the claimed $\mathbb{P}$. This typically comes in two forms. One form is identifying the limitations to the claimed $\mathbb{P}$, and the other is identifying alternative $\mathbb{P}'$ that demonstrate that the claimed $\mathbb{P}$ is not unique and that alternatives refute the accusation. To the extent that these are both done, it benefits the accused.

Identifying refutations of P

To the extent that claims in $\mathbb{P}$ can be refuted, this is the strongest argument against those claims. While science might, in some cases, assert that a single refutation destroys such a claim, this is not always true in the legal system. Because $m \epsilon \mathbb{P}$ are not all purely mathematical in nature, and because all $\mathbb{C} \rightarrow \mathbb{E}$ are not precise, there may be cases when a refutation is imperfect. In addition, since such information is generally shown to triers of fact, a single refutation may not be adequately convincing, even if, as a scientific claim, it is compelling. A refutation might be stated something like this:

> "Based on [basis], the claim that [$\mathbb{C} \rightarrow \mathbb{E}$] for [m] cannot be true. Based on the fact that [$\mathbb{P}$] depends on [m], [Other

party]'s claims are inconsistent with the [traces and events] and, thus these claims cannot be and are not true."

Of course this is not always the case, and care should be exercised in going too far in such a refutation when there are alternatives available to the other side. Leaving challenges at the level of the inconsistency may be adequate in many cases.

Demonstrating alternative $\mathbb{P}$

A second, and less compelling course to countering claims is to identify alternatives that might just as well be true. For example, even though traces are consistent with $\mathbb{C}\rightarrow\mathbb{E}$, they may also be consistent with an alternative $\mathbb{C}'\rightarrow\mathbb{E}'$, in which the accused is innocent. A demonstration of alternatives might be called out something like this:

"[Other party] claims [$\mathbb{C}\rightarrow\mathbb{E}$], but many other possibilities exist and are consistent with [the relevant traces and events]. For example, and without limit:

[$\mathbb{C}'\rightarrow\mathbb{E}'$] (e.g., Joe's brother was present in the room and had access to the same stuff)

[$\mathbb{C}''\rightarrow\mathbb{E}''$] (e.g., Joe's wife, who is suing for divorce, and who previously …)

..."

Identifying such alternative possibilities, even though this is not as strong as refutation of $\mathbb{P}$, may be strong enough to cause a judge or jury to become unconvinced that the standard of proof has been met for the matter at hand. The strength of such alternatives, presumably, increases as there are more and more convincing alternative paths shown. Such paths tend to be strengthened when they are supported by traces and events, and when they sound reasonable to the trier of fact. For example, when such alternatives include motives for 3rd parties, traces indicative of 3rd parties, traces showing a lack of consistency with the other party's claims, and similar supporting details, they become far stronger and, ultimately, may be as compelling as refutations in the minds of the triers of fact.

Identifying limitations in $\mathbb{P}$

The least compelling, but still often viable and effective approach to countering claims, is to emphasize limitations in $\mathbb{P}$ as presented by the other party. This includes identifying limits on granularity and thoroughness, cases where traces or events were not found so consistency could not be demonstrated, and cases where traces found did not demonstrate consistency, even if they did not show inconsistency. These may be enumerated in a statement like this:

> "[Other party] claims $\mathbb{P}$, including [$\mathbb{C}\rightarrow\mathbb{E}$], ..., however, and without limit (and as/if appropriate):
>
> [Other party] fails to provide a basis for the claim that [$\mathbb{C}\rightarrow\mathbb{E}$]
>
> [Other party] fails to consider low-level mechanisms that might refute the claim that [$\mathbb{C}\rightarrow\mathbb{E}$]
>
> [Other party] has not shown traces consistent with the claim that [$\mathbb{C}\rightarrow\mathbb{E}$] (despite having traces that (might/would) so indicate if [$\mathbb{C}\rightarrow\mathbb{E}$] were in fact the case).
>
> [Other party] fails to identify and examine additional mechanisms such as [list some of them] necessary for [path $\mathbb{P}$] to actually take place.
>
> [Other party] only performed a limited set of procedure(s) [P] and in failing to perform procedure(s) [P', ...] did not account for [traces and events] that might have demonstrated [Party's] innocence."

There may be many such limitations identified, and to the extent that these limitations are considered substantive by and meaningful to the trier of fact, they may carry enough weight to sway the trier of fact below the threshold required to find the accused guilty.

An applied approach to forensic analysis of control

In examining the issues of control associated with causality, the overarching analysis methodology described above is outlined here. Elements of the analytical framework are identified by number sequences within the outline provided (e.g., 1.1.2.1.1 is “Uncovered path”). For notational purposes, we will identify such elements by placing them in curly brackets (e.g., {1.1.3.1} indicates “Acts within the control envelope” for a “general purpose mechanism in normal

use"). "~" is used for "not shown" (e.g., {~1.1}) and ! for refutation (e.g., {!1.1.4.1.1} → shown false).[333] Attribution to source is also required to demonstrate a violation by any particular party, if and only if control has been established.

0 No control (evidence refutes violation) ⊗
- 0.1 No syntax to express identified intent (the act is thus outside the syntactic control envelope) +
- 0.2 No authority to carry out intent (the act is thus outside the semantic control envelope)

1 Control (evidence supporting violation)
- 1.1 Direct +
- 1.1.1 Special purpose mechanism in normal use ⊗
  - 1.1.1.1 Acts within the control envelope *
  - 1.1.1.2 Traces evidence use of syntax *
  - 1.1.1.3 Traces evidence semantic effect
- 1.1.2 Special purpose mechanism exceeded ⊗
  - 1.1.2.1 Evidence mechanism(s) to exceed *
  - 1.1.2.1.1 Uncovered path +
  - 1.1.2.1.2 Exploited weakness
  - 1.1.2.2 Traces indicate envelope exceeded *
  - 1.1.2.3 Acts in recursive control envelope *
  - 1.1.2.4 Evaluate for enclosing envelope
- 1.1.3 General purpose mechanism in normal use ⊗
  - 1.1.3.1 Acts within the control envelope *
  - 1.1.3.2 Traces evidence use of syntax *
  - 1.1.3.3 Traces evidence semantic effect
- 1.1.4 General purpose mechanism exceeded
  - 1.1.4.1 Evidence mechanism(s) to exceed *
  - 1.1.4.1.1 Uncovered path +
  - 1.1.4.1.2 Exploited weakness
  - 1.1.4.2 Traces show envelope exceeded *
  - 1.1.4.3 Acts in recursive control envelope *
  - 1.1.4.4 Evaluate for enclosing envelope
- 1.2 Indirect
- 1.2.1 Indirect mechanism identified as within a new control envelope

AND 1.2.2 Apply above analysis in new envelope

333 F. Cohen, "A Case Study in Forensic Analysis of Control", JDFSL, 2011.

Based on an assumption, we might reasonably conclude that for an identified root user, {1.1.3} applies to any activity that root user would perform on their own system. {1.1.3.1} A stronger demonstration of a particular act (e.g., deleting a file) would be the presence of traces indicative of a particular command to perform that act (e.g., “rm /etc/passwd” in the log of console commands), {1.1.3.2} and traces of the results of that command that are consistent with that act. (e.g., no such file present) {1.1.3.3} We might then reasonably come to a logical conclusion such as that {1.1.3.1}*{1.1.3.2}*{1.1.3.1}→{1.1.3}→{1.1}→{1} and that the actor logged in as “root” apparently had control and performed the act.

In such cases the weight of the accumulated evidence associated with traces and events, assuming relevance, reliability, and other similar things could be shown, would presumably be adequate to be presented to the trier of fact. Similar evidence by the opposing side would have a similar threshold of admissibility and the trier of fact would be left to decide the issues. For example, if traces normally present are missing, this would make the reliability of the claims more dubious, particularly if the reconstruction produced them and the traces from the other party did not.

Suppose, for example, that the accused individual claims that they were elsewhere at the time and that facility logs kept by an independent special purpose system indicate that the individual was elsewhere at the time and no such records indicate that they were present at the location of the console of the computer at issue at the time. The individual might reasonably claim that the facility logs showed they were elsewhere, and since that they don't have the ability to produce false logs from the normal user interface ({~1.1.1.1}→{~1.1.1}), and no evidence was present that those logs were acting improperly ({~1.1.2.1}*{~1.1.2.2}→{~1.1.2}). Since this is a special purpose system ({~1.1.3}*{~1.1.4}), it is reasonable to conclude that the accused had no direct control over the facility logs ({~1.1.1}*{~1.1.2}*{~1.1.3}*{~1.1.4}→{~1.1}).

Presumably, this would be admitted to demonstrate the innocence of the accused. But the accuser might assert that there are ways to get around such mechanisms, like trading badges with a partner. {1.2} In this scenario, a new control envelope is present {1.2.1} and the above analysis can then be repeated within this new control

envelope. In the new claimed control envelope, the facility log operates normally but the mechanism's intended use is asserted to be exceeded {1.1.2} by an exploited weakness {1.1.2.2} by the accused and their alleged partner. Traces are present of the excess {1.1.2.2} in the presence of the alleged partner in the location with the console the acts are within the recursive control envelope for the partner and the accused ({1.1.1.1}*{1.1.1.2}*{1.1.1.3}→{1.1.1} working together → {1.1.2.3}*{1.1.2.4}), and thus {1.1.1}→{1.1} (indirect)→{1.2}→{1}, and the violation is supportable by available traces. However, the “evidence” part of {1.1.2.1} is weak lacking evidence of partnership or acts.

The reality of complex attributions

Complex attributions may involve large bodies of traces and events that combine to provide the examination result.[334] As an example, in one case,[335] an examiner identified a complex header sequence associated with a party and found the sequence in 64 of 200,000+ messages. Tools revealed that sequences and content of headers were almost identical. This supported extending attribution to these 64 messages, which when examined, showed an independent linkage back to the same party. 63 of these 64 messages were also part of an independently identified set asserted attributable to the behavior of the party and the 64th was self-asserted to have been posted by the party. But that technical component was only the beginning of the overall attribution. One of the three overall attributions was summarized as [details redacted]:

GA appears to have taken confidential and/or privileged emails:

- Two specific emails at issue and many other emails were identified by XXX as privileged and/or confidential.
- All parties to the specific emails indicated they did not reveal anything about the emails to anyone in relevant time frames.

334 F. Cohen, "Attribution of messages to sources in digital forensics cases", HICSS-43, Jan 7, 2010.

335 Susan Polgar v. United States of America Chess Federation, inc., C.A. NO. 5-08CV0169-C in the United States District Court - Northern District of Texas - Lubbock Division.

- Unauthorized accesses to XXX's email account containing the emails at issue, were made after the emails were stored in XXX's account, and before they were publicly released.
- At least 100 unauthorized access attempts to XXX's email, some or all apparently successful, including those related to the two identified emails, came from IP addresses that:
  - (1) were used in comparable time frames to post to the USCF online forum under GA's user name, and/or
  - (2) were used in comparable time frames to post to the usenet forums under GA's identity, and/or
  - (3) were used from an Anonymizer account GA held and that was used from IP addresses (a) assigned to GA by Comcast, (b) used to make postings to the USCF online forum from GA's USCF account, and/or (c) used to make newsgroup postings under GA's identity.
- The two specific emails were then first publicly released in emails sent by SP's account, for whom GA worked on a volunteer basis, and whose Web site GA operated. ...

GA appears to have then released taken emails:

- The sequence of events with respect to the "U" Web site is summarized as follows:
  - 2008-07-30 at 04:10:13 GMT: The account "V" was created on or about 2008-07-30 at 04:10:13 GMT from an IP address [234]. This is in the IP address range of other addresses associated with Anonymizer and under the control of NTT America.
  - 2008-07-31 at 02:00:33.739 GMT: The "U" blog was created using the "V" Yahoo! account for ownership identification, and accessed at that time from Anonymizer IP address [249]. Access through Anonymizer at this time was undertaken by the user identified as GA through Anonymizer records and from the IP address [165].
  - 2008-07-31 from 02:02 to 03:57 GMT: The "U" blog was accessed repeatedly from IP address [249]. Access through Anonymizer at this time was undertaken by the

user identified as GA through Anonymizer records and from IP address [165].

- 2008-07-31 at 08:02:00 GMT: The "U" blog was accessed two times from IP address [50].
- 2008-08-06 at 08:59:40 GMT: IP address [229] was used, to obtain unauthorized access to the XXX@Y email account, by an individual identified as GA according to the Anonymizer logs and analysis.
- 2008-08-07 at 04:43:00 GMT: The same IP address, [229] was used to access the blog "U"
- 2008-08-08 at 09:13:10 GMT: The IP address [179] was used to obtain unauthorized access to the XXX@Y email account, by an individual identified as GA according to the Anonymizer logs and analysis.
- 2008-08-08 at 21:52:39, 22:26:08, and 23:53:19 GMT: The same IP address, [179] was used to access the "V" Yahoo! account.
- 2008-08-31 at 13:22:48 GMT: The IP address [11] was used to access the blog "U". This is also an IP address previously used for postings to RGCP and RGCM under GA's identity, and an IP address at the University of W from an area where GA works.

- These comprise all of the sessions where postings and activities to control the "U" blog were recorded by Google, the operator of this site. SP's attorney identified this as the site where she came to first possess the information. The printout provided thereby appears to show that this site had this information, but postdates SP's release of that content.

Ignoring the particulars of this matter, the attribution process shown involves combinations of events and traces, there are many sources of information, there are redundant paths from cause to effect, statements and actions taken by parties as well as DFE are involved in overall attribution, and results are couched in terms of what appears to be the case, and not as absolutes.

Logical fallacies in attribution

This brings us back to the common human fallacies discussed earlier, and it is worth reexamining them in light of the challenges facing attribution. Table 7.7 gives examples of how fallacies might be reflected in attribution.

Fallacy Mechanism	*Example*
Effects should resemble their causes.	
- instances should resemble their categories	If the traces indicate use of complex methods, an expert must have done it. (Of course experts write a lot of software that is used by non-experts, and one person's expert may be another person's amateur.)
- like resembles like	The pattern of activities by person A in system B look like the pattern of activities in system C, so person A must have made them. (Similarity of effects does not imply the similarity of causes)
- tendency toward oversimplification	If A sent it and B received it, A must have sent it to B. (see the discussion above)
- Occam's Razor	If a steganographic program P is present and file F has steganography in it that P can create, P created the steganographic content in F. (A different program might have caused the same result.)
- black and white	The traces were either caused by Joe or they weren't. (Perhaps they were partially caused by Joe and partially caused by some other actor.)
- rule of 3s	Either Joe sent it, Mary sent it, or neither of them did. (What if both of them did?)
The misperception of random events	

Fallacy Mechanism	*Example*
- the clustering illusion	Each of the 35 factors are within 25% of the average behavior for Joe and nobody else. (This may simply be a random result.)
- over-application of representativeness	We examined a sample of 5,000 files for time and date stamps, and all of the traces were consistent with the events. (Many small computers today have more than a million files present, and a sample of this size may have little or no statistical value, depending on the way sampling was done and whether statistics even applies to the issue.)
- misperceptions of random dispersions	File accesses were highest at times when Mary was present that week. (This may have nothing at all to do with Mary but may simply be a result of other unrelated phenomena.)
- the creation of casual theories	The files contained special characters so that Mike could find them easily. (Or they could be there so Jane could use them, or they could be there because of the program used, or for any other sort of reason.)
- the regression fallacy	The system seemed much slower when the new program was running. (Or it could just be that things were slower that week for other reasons.)
Misinterpretation of incomplete or unrepresentative data	

Fallacy Mechanism	*Example*
- the excessive impact of confirmatory information	We examined traces of system accesses and CPU usage and both confirmed that the slowdowns were seen when Jerry was there. And that was true 4 days out of 5 that week. (Isn't the 5th day a refutation that disproves the theory?)
- the tendency to seek confirmatory data	In searching for evidence that Jerry did it, we found 17 of his files with suspicious sounding names. (How many would be suspicious sounding if not being sought for that purpose or if being sought for a different purpose?)
- the problem of hidden or absent data	The identified traces and events show that A sent the items to B. (Until we find out the next piece of information that makes attribution questionable at best.)
- self-fulfilling prophecies	We looked for evidence of steganography and found 350 files that appear to have unusual characteristics. (The notion that there will be unusual characteristics leads to the search for them, and the search will continue until something is found to call unusual.)
The biased evaluation of ambiguous and inconsistent data	
- ambiguous information is interpreted in context	We found that 65% of the file change time stamps were consistent with the times Jerry was working. (But maybe that is also true of Mary and Paul.)
- unambiguous data is shaded	We ignored the changes when Jerry wasn't working because he wasn't there. (The refutation of Jerry as the cause is ignored because it is inconsistent with the theory.)

Fallacy Mechanism	*Example*
- multiple endpoints	Either Jerry would come back via the network or from a local console, and sure enough, we detected network activity. (No matter what the events, Jerry will somehow be found responsible.)
- confirmations and non-confirmations	We found that some of the changes happened when Jerry wasn't present, but he might have been remotely logged in. (Any excuse will do to ignore refutations.)
- focused and unfocused expectations	We predicted that at some point Jerry would do something else, and sure enough, a week later, a similar activity started from somewhere else. (No matter how long it takes, we will wait for something we can claim shows that Jerry did it.)
Outcome asymmetries and one-sided events	
- hedonic asymmetries	The telling traces were the log files that clearly showed that Mary wasn't logged in. (Why is this telling as opposed to all of the other traces?)
- pattern asymmetries	I remember that by 9:19 the activity was started every day, and Jerry works in room 919. (Was it also started by 9:16 every day? That isn't Jerry's room number!)
- definitional asymmetries	Performance got slower and slower until Jerry finally got bored and went on to something else. (The definition of "Jerry got bored" is undefined and unmeasured except that performance got better.)

Fallacy Mechanism	*Example*
- base rate departures	The messages sent by Jerry slowed our server. (No comparable data is available without those messages to show that the server would otherwise have been faster.)
Motivational determinants of belief	
- empirical support for the wish to believe	We suspected Jerry from the start, and the more evidence we looked for the more we found. (Of course this could also be true of Mary if you suspected her.)
- mechanisms of self-serving beliefs	Plaintiff believes that the use of Jerry's user ID and password from the console proves that Jerry was there. Defendant believes that the lack of building entry records for Jerry proves that Jerry wasn't there. (If you want to believe you will find a reason to do so. If you don't want to believe, you will find a reason not to.)
- optimistic self-assessment	I know what Jerry's programs look like, and this is one of Jerry's programs. (The examiner believes that they can look at a program and tell the author.)
The biasing effect of second hand information	
- sharpening and leveling	It is clear that Jerry logged in at the console, even if he managed to avoid the building entry system. (The login at the entry console is exaggerated (sharpened) while the lack of entry to the building is underplayed (leveled).)
- the corrupting effect of increasingly indirect evidence	The game of 'telephone' is a great example - hearsay evidence is excluded for this reason.

Fallacy Mechanism	*Example*
- telling a good story	In the old days, what Jerry did was called "a salami attack" because they took it one slice at a time, but now we just call it fraud. (The story from olden days makes it seem more interesting and memorable.)
- distortions in the name of informativeness	A few years ago, someone broke all of the entry and exit security, so Jerry must have done the same thing. (When in reality, 3 years earlier, a security test found some ways to access the facility that were fixed after the results of the test were known.)
- distortions in the name of entertainment	I caught Jerry red handed. (When the best that is really available are a few traces that are inconsistent with events.)
- distortions in the name of self interest	The lack of records of Jerry's entry proves that he was not in the building when the alleged activities took place. (So says Jerry's lawyer.)
- distortions due to plausibility	More than 70% of all spam comes from companies like this. (How exactly can this be accurately measured, what exactly is a "company like this", and by what definition of "spam"?)
Exaggerated impressions of social support	
- social projection and the false consensus effect	Most experts would agree that the use of a User ID and password clearly show the presence of the user. (What study of what sorts of experts was used to come to this conclusion?)

Fallacy Mechanism	*Example*
- inadequate feedback from others.	I asked experts on several forums about this issue, and not one of them had even a single counterexample. (Many of them might have, but perhaps they didn't feel like publicly embarrassing you.)

Table 7.7 Logical fallacies common in humans applied to attribution

Clearly, there is a lot of potential for logical fallacies in attribution. But it is a mistake to assume that, just because questions can be asked regarding these sorts of fallacies, that implies that the attributions are not true or valid. It would be a fallacy to conclude that the presence of a fallacy means that the thing about which the fallacy applies is not true, just because the method by which it was shown is problematic.

Questions

1. Given that it is impossible to perfectly attribute traces to human actors with DFE alone, what is the best the examiner can hope for in terms of attribution in the legal sense?
2. Given the seeming incompatibility of statistics with the digital world and the extreme nonlinearity of digital systems, what are the cases when statistics will be more probative than prejudicial in digital forensics?
3. Given that forward execution of FSMs is highly causal in nature, in an ideal circumstance where all input sequences are available, can perfect causation be attained for a digital system? If not, why not? If so, explain how.
4. Give an example of how sensors from unrelated systems and mechanisms might be used to help lock down an attribution. Assume that the overall environment is a modern office building with automated badging systems, controlled access to floors and offices, heat sensors for lighting, air temperature controls, power controls, and that all activities of all sensors are recorded and available in traces for the examination.
5. Given the example from question 4, how would a competent examiner with strong evidence that the accused individual was not present be able to counter the attribution? What challenges would be used?
6. Using the detailed results indicated in the various papers cited for authentication of individuals, can you identify a probability that an individual identified by these methods is who the methods assert that they are? If so, pick two methods and assume that both of them indicate the same individual. What would the probability then be?
7. Given your answer for question 6, assume that you reported that probability, and that after your report was filed, a retinal scan was also identified, and that this scan showed that the suspect was not the individual present. How would this change the probability?

8. Based on Table 7.1, what is your general conclusion about level 3 attribution of network behaviors today?
9. Using Table 7.2 as a baseline, and assuming that ingress filtering is used, how useful under what circumstances is this for level 1 attribution?
10. Using Table 7.3, assuming a suspect is using anonymization and that logs and snapshots are available as traces, is there a way to attribute actions to actors? Identify the circumstances under which this will be effective and in which it will not be effective, and give a metric for how effective it will be.
11. Assuming that TCG TPM mechanisms are in use in the computers of interest and that they provide information relating to the computers involved in a series of activities, what sorts of attribution could be made, and how certain would those attributions be?
12. The use of encryption and digital signatures is often touted by some in the cryptographic community as so hard to forge that they provide definitive proof of origins of content. Suppose there is a cryptographic proof that your client digitally signed a document. Would you simply tell the lawyer to stipulate it as the truth? If not, what examination would you perform to try to show that the signature is a forgery? What results would indicate that it is a forgery?
13. Given that an HTML file appears in a browser cache area, list ten examples of how this could happen without the browser having left it there. How would you use other traces to confirm or refute these attributions? Would the results of these examinations show that the file was caused by the browser? If not, how would you show this attribution?
14. Given the information physics limits of attribution, review your answer to the previous question and identify weaknesses in your method for doing attribution. How would you use traces to bolster the case in light of these potential weaknesses?

15. Correlation is not causality, but ordering is necessary for causation in the forward-moving digital world we normally consider. Given the uncertainties associated with time in the digital world, what conditions on traces related to timing would you require as mandatory for asserting causality of one event by another?
16. Given the limits on damages associated with trespass, what trespass damages could be reasonably asserted as a result of someone logging into a user account with a stolen password and using that account to look at publicly available information over the Web? How would traces be used to demonstrate that damage?
17. Suppose a series of emails sent by a known party were the subject of a claim in a civil case that the emails degraded communications by consuming bandwidth, and that the only traces offered as evidence are the traces of the emails themselves. Using only the "Received:" headers, could you attribute bandwidth deprivation to the emails? If so, how? If not, why not?
18. Using Table 7.5 to codify particulars for the last question, identify the likely information that could be derived from the traces in the email headers, how damage amounts could be associated with them, and the basis for attribution in forensic properties.
19. It is true that mens rae and attribution go together? If so, explain why and how. If not, explain why not.
20. Find an example of attribution from over the Internet, and using the logical fallacies identified in Table 7.7, evaluate that attribution in terms of the fallacies.
21. Given all of the problems involved in attribution, what is the most definitive statement that can realistically be made regarding the attribution of acts to humans based only on traces? If a more definitive statement is asserted, how is it most certainly not accurate or definitive, and how can it be reasonably challenged?

8 Reconstruction

In this book, the experimental component of DFE examination is called reconstruction. At the end of the day, all of the analysis, interpretation, and attribution in the world has to stand up to the scrutiny of testability in order to be considered scientific. This testability criteria of science demands that, in order for a hypothesis to be confirmed or refuted, an experimental method must be devised with predicted results. If those predicted results fail to be realized, then the theory is refuted. If those results are realized, this may act as a confirmation of the theory, but it does not "prove" the theory to be true unless the experiments can exhaust all of the possibilities that the theory addresses.[336] The approach to reconstruction discussed here is the one identified in[337] and discussed in some detail in 2008.[338]

Reconstruction as driving time backwards

There is actually another use of the term reconstruction with regard to DFE, and that is the attempt to run time backwards.[339] In Carrier's dissertation, reconstruction is identified as, in essence, running time backwards to determine the previous inputs and states that must have or could have produced the traces. From a standpoint of information physics, there are several hurdles to get over to realize such an approach. For example, supposed we execute instructions: "Input x; Input y; print x+y;". From the result (x+y) how can we know what x and y were inputs? In broader terms, the aspects of information physics identified in Table 8.1 are problematic for the reasons given.

Digital World	Carrier reconstruction problems
Finite time granularity (the clock)	Time reversal has time ranges
Exact copies, original intact	Inability to tell what was original

336 K. Popper, The Logic of Scientific Discovery (1959), Hutchins and Company, London. ISBN10: 0415278449.

337 F. Cohen, "Digital Crime Scene Reconstruction", Presentation at the 2006 DoD CyberCrime Summit, Jan 12, 2006.

338 F. Cohen, “Challenges to Digital Forensic Evidence”, ASP Press, 2008.

339 B. Carrier, "A Hypothesis Based Approach to Digital Forensic Investigation." PhD Dissertation; Purdue University; May, 2006.

Digital World	Carrier reconstruction problems
Theft without direct loss	Theft may not be identifiable
Finite (fast) rate of movement	Travel time and jitter produce uncertainty in reverse time
Homing sequences may exist	Unique reverse time through homing states is impossible
Forward time perfect prediction	Allows limited validation
Backward time non-unique	Expanding sets of histories likely
Digital space converges in time	Convergence makes prior states non-unique
The results are always bits	Limited maximum granularity to examine for inputs
Results are always "Exact"	Predictions refutable but not always confirmable
Time is a partial ordering	Reverse time may have many possible orderings
Errors accumulate	Reversing possible errors may lead to history expansion
Representation limits accuracy	Reversal cannot get back lost accuracy
Precision may exceed accuracy	Excess precision may allow more accurate reverse time
Forgery can be perfect	Reverse time cannot detect all forgeries and may give false reverse sequences
DFE is almost always latent	Tools will have to be used for reverse time and tool errors may produce expanding reverse time errors
DFE is circumstantial	All possible circumstances cannot be explored in reverse time.

Digital World	Carrier reconstruction problems
DFE can show consistency or inconsistency only	Inconsistent traces lead to very expansive time reversal
Content perfectly compressible	Imperfect compression results in large expansions of time history
Digital signatures, fingerprints, etc. generated from content	Reverse time is (intentionally) high complexity and unlimited (except for time taken) non-unique histories result.
Content meaning is dictated by context	Expansive number of reverse time contexts are possible
Context tends to be global and dramatically changes meaning	Expansive number of reverse time "meanings"
FSMs come to a conclusion	Expansive possible sources of conclusions
Cognitive limits from program	Reversal through cognitive errors produces large numbers of possible histories
Hardware fault models from computer engineering	Fault models allow large numbers of alternative histories
Near perfect virtualization and simulation possible	Virtualized environments may yield enormous and indistinguishable histories
Many nearly or equivalent FSMs	FSMs with equivalent outputs may have many different inputs and states
Undecidable problems	Much of time reversal is worse than undecidable, it is inherently not differentiable
Computational complexity limits computations	Time and space complexity of time reversal extremely high
Consistency is guaranteed	Reverse inconsistency is almost always guaranteed

Digital World	Carrier reconstruction problems
Completeness is guaranteed	Reverse completeness may not be possible
Complexity-based designs	Forward complexity may lead to greater reverse complexity
Fault tolerance by design	Fault tolerance causes reverse time expansion of possibilities
Discontinuous space	Ranges of history values result
Discontinuous time	Ranges of history values result
Minor differences amplified near discontinuities	Reverse time indistinguishable ranges ∀ values ∀ inputs
Major differences suppressed away from discontinuities	Reverse time indistinguishable ranges ∀ values ∀ inputs
Identical use of an interface may produce different results	Very high numbers of different external states produce inputs
Ordering may be reversed	Ordering reversal yields exponential expansion of possible originating sequences
Value sorts may be reversed	Reversal means exponential expansion of value ranges
Actuate-sensor loop errors	Reversal through errors yields high numbers of expansions
Sensors/ actuators limited in physical properties	Any unobserved physical property can be anything in time reversal

Table 8.1 - Information physics and Carrier reconstruction

While proofs of these conclusions are not provided here and likely is unavailable in the literature, these hypothesized results are very likely to be the case for reverse time reconstruction. But despite these reverse time challenges, class sets of previous states and inputs for limited numbers of FSMs may be differentiable and this may be sufficient for some cases.

Reconstruction as an experimental approach

The perspective of this book is that, as a science, DFE examination needs an experimental component, and that component is called reconstruction. Following the basic tenants of scientific investigation, an experiment about an infinite set (or in the case of DFE histories, a very large finite one) cannot prove a theory, it can only confirm or refute the theory. But in most cases, DFE examination is not working so much with the underlying theories of the science. Rather, in most cases,

> **Reconstruction is used to test hypotheses about the particular case. It can confirm, refute, or be unrevealing.**

The difference between a hypothesis and a theory in science is that a theory is normally a general principle that applies to an enormous breadth of cases and is usually well tested by many experiments over a long period of time. To the extent that a previously accepted scientific theory is wrong, it is almost always a very good approximation to right in almost all cases of certain types, and the theory that replaces it shows why the old theory was wrong, by how much, and in which cases. A hypothesis is typically closer to a guess in the sense that it is an attempt to explain a particular trace and set of events. It is far more speculative than a theory. Hypotheses are thrown up on the spur of the moment and tested using the theoretical underpinnings and experiments. Many of the hypotheses thrown up are refuted in reconstruction. Reconstruction can also be unrevealing with respect to a hypothesis.

Some word usage and definitions

Reconstruction is about creating hypotheses based on traces and events, producing traces from those hypotheses, and comparing the resulting examiner-produced traces to the traces produced as part of the potential or actual evidence in a legal matter. Word conflicts may arise unless terms are defined.

We will call a trace produced by the examiner as a result of reconstruction a constructed trace, or a C-trace. An original trace produced and asserted as original writing or otherwise valid evidence in the legal matter, or any other trace that is at issue in the matter at hand, will be called an original trace, an O-trace, or simply a trace.

The use of the term "event" in discussing reconstruction in this book is consistent with its use throughout the book and with the model of examination presented earlier. But the usage may seem loose in that the discussion of histories produced by reconstruction is done in terms of event sequences and similar usage. It may be instructive to recall that events include hypotheses, documents, statements, rulings, and other things that are not traces, and the consistency of which are weighed with traces to produce analytical results, interpretations, attributions, and reconstructions. While the usage may seem loose at times, it will be helpful to keep things in this context while reading in order to retain clarity.

Classes of inputs, states, outputs, and traces, are sets of similar items of each sort characterized within an "envelope" of items, or "class sets". The envelope is defined by the characterization used. For example, assume a reconstruction of events related to a login, where the goal of the reconstruction is to test the hypothesis that:

> *2 failed login attempts under the same user identity within 60 seconds do not cause a trace starting with "Login failure" and including the user identity entered in the identified audit logs within 600 seconds of the failed attempts, but 3 failed login attempts under the same user identity within a 60 seconds do cause such a trace.*

The resulting envelope of inputs includes all user identity strings, all 60-second sequential periods of time, and all passwords that do not properly authenticate each of those user identities. The envelope of states include all machine states. The envelope of outputs other than those producing traces is all outputs. The envelope of traces that refute the hypothesis includes all traces that do not include the identified indicator of the failed login attempts within the identified audit logs within 600 seconds after 3 attempts or do contain such traces after 2 or fewer attempts. Of course the reconstruction might fail to produce such a trace in a case where there is inadequate disk space, and an additional condition may then be placed on the envelope. But we don't know this until it is tested, because the system, for example, might refuse to allow any login attempts when there is inadequate disk space to store audit trails of this sort.

Consistency of a trace with an envelope indicates that the trace is an element of the set defined by the envelope, and inconsistency of a trace with an envelope is the non-containment of that trace within that envelope. Consistency of a C-trace with an O-trace either indicates that the C-trace is identical to the O-trace or that the C-trace and the O-trace are contained within the same pre-defined envelope.

Forward computation is a computation that proceeds in the normal manner of the FSM under test, from initial state, taking inputs, producing subsequent states, and producing outputs.

The notion of a history of a machine is the classic characterization of the sequence of all inputs, states, and outputs of the FSM over time. When identifying multiple histories, the notion is the set of all histories relevant to the issue at hand.

Forward reconstruction defined

Forward reconstruction, or simply reconstruction, attempts to recreate conditions that may have existed at some point in time with the objective of causing the recreation to behave as the original might have behaved in all materially relevant ways.

> If the results of this reconstruction generates C-traces that are **within the envelope** equivalent to the O-trace, then the C-trace and reconstruction are said to be **consistent** with the O-trace.
>
> If the result of this reconstruction generates C-traces that are **not within the envelope** equivalent to the O-trace, then the C-trace and reconstruction are said to be **inconsistent** with the O-trace.

This can be done either to try to find a reconstruction that produces consistent C-traces with the O-traces, or to test a hypothesis by reproducing the hypothesizes situation to see whether the resulting C-traces are consistent with the O-traces.

Of course this notion introduces many challenges. The challenges include, without limit, (1) the initial state of a system is almost never known, so it has to be hypothesized, and when a reconstruction fails to produce consistent C-traces, it is not known whether the initial state hypothesis or other hypotheses are falsified; (2) the

definition of "material" is not clear, and what is or is not "material" may be purely a matter of judgment; (3) the granularity and precision requirements of the reconstruction have to be determined based on the materiality issue; (4) the definition of what is or is not consistent with regard to O-traces and C-traces has to be identified in terms of what is sought; and (5) often the O-traces and events available are a limited subset of the things required for reconstruction, so the rest of the initial state, other inputs, and FSMs used to do the reconstruction, constitute hypotheses as well.

As an example, a reconstruction of a message handling application receiving messages and forwarding them onward may involve a set of hypotheses (read educated guesses) about the hardware, the operating system, the message processing application, the settings of all of these things, and other things occurring in the computing environment. Depending on what the issues are in the legal matter, different parts of these things may be important. For example, if the issue has to do with whether a break-in to the computer caused a record of a message to have been corrupted, this is very different from determining whether a particular message server adds a particular sort of header with particular characteristics. In the former case many configuration-related issues across all applications and configurations involved may impact the outcome, but in the latter case, it may be enough to show that the application adds a specific header with a particular syntax in the default configuration.

When the result of a reconstruction is challenged, it may be challenged based on information available, or it may be challenged based on an incorrect assumption. For example, in the message processing case, the opposing side might indicate that the wrong message processing software was used in the reconstruction. The fact that the other side refused to provide the details of what software was in use and that there are other O-traces or events that are consistent with the processing software as reconstructed and inconsistent with the newly claimed event will not prevent the challenge from being used. Another reconstruction can be undertaken using the newly added information, the inconsistent claim can be challenged, or this can be ignored, depending on the specifics of the legal matter and the schedule.

What can be easily tested by reconstruction and how fast

The easiest sorts of tests that can be done through reconstruction tend to be determining how something normally operates in terms of the C-traces it produces. This is simply a matter of constructing a system of a similar sort and operating it. Assuming that, in normal operation, the relevant events occur at a rate of *x* events of interest per *y* units of time, the reconstruction should reproduce the first indicative trace in about *y/x* units of time. If the rate is statistical in nature, or in other words, if the causes of the events of interest are not known precisely or easily recreated, then some variance will be expected, and this must be considered in the time to wait before a confirmation.

The lack of a confirmation within the defined time frame would then constitute a refutation, but "A refutation of what?" remains a question. Is it a refutation of the theory of the case, of a hypothesis about arrival times of events, the many hypotheses about the system configuration, or what? The examiner has to identify which and determine whether and to what extent additional experiments may resolve the issue. If the C-trace is statistical in nature, or in other words, if the exact mechanisms are not known or producible, this may lead to reasonable limits on the certainty attained by a given number of experiments with the reconstruction. The desired confidence level may then be used to drive the number of experiments performed, or other statistical or analytical methods may apply. These factors combined with the schedule drive the examination process and analysis and interpretation of results.

Precision issues and prediction prior to experimentation

The precision of the reconstruction also drives the precision with which the C-traces may be expected to be consistent with the O-traces in order to be considered a confirmation or refutation. For example, if the reconstruction is designed to identify the presence or absence of a particular set of symbols in a C-trace, the ordering of traces in the C-trace will be irrelevant to the issue of whether a consistent C-trace was produced. But if the question is one of ordering, then the ordering will be important, and the partially ordered set that constitutes consistency as opposed to the remaining orderings, which are inconsistent, must be defined in

advance of the experiment in order for the prediction of the theory to be confirmed or refuted.

From a scientific standpoint, this is very important, as it eliminates the common fallacies related to the biased evaluation of ambiguous and inconsistent data and reduces the misinterpretation of incomplete or unrepresentative data. The common fallacies of multiple endpoints and unfocused expectations are excluded by the predictive approach. If the precise limits on confirmatory orderings are not defined in advance, then the experimental results may end up skewing decisions related to what constitutes a confirming ordering. In cases where a definitive answer is not available, a formulaic approach may be used to measure the extent of confirmation from orderings, where there are cases when one set of orderings confirms, another refutes, and a third set is not revealing or only partially confirming or refuting.

When timing is an issue in the case, the constraints on the differences between O-traces and C-traces should similarly be defined in advance, and the ranges of values for the metric used in measuring results should be defined in advance, along with conclusions in terms of confirmation or refutation associated with the different measurable outcomes. Again, precision may become an issue as the difference between the C-trace and the O-trace approaches the boundary between confirmation and refutation, and a region for uncertain results will necessarily apply based on the limits of precision and certainty about the implications of outcomes.

Repeatability of reconstruction results

Repeatability is also a hallmark of scientific approaches. To the extent that repetition is feasible with reconstruction and within the confines of the schedule, repetition should be undertaken in experiments of all sorts. Repeatability also implies proper documentation, which normally implies contemporaneous notes. Such notes are problematic in legal matters because lawyers do not like notes that the other side can force to be disclosed during discovery. Such notes can often be easily misunderstood, misinterpreted, or used to show the imperfect nature of the scientific method and the scientists who apply it. In the arena of DFE examination, computers greatly ease the burden of taking

notes by automatically taking them - hence the traces produced in cases. However, it should not be an exercise in DFE examination to find out what the examiner on the other side did to come to the results they produced. Results should be readily reproduced or verified by a competent examiner on the other side by repetition of the documented methods. To the extent that this is not the case, the results and the methods used should be challenged if they are important to the case or to the credibility of the examiner.

When is reconstruction not needed or revealing?

There are some common DFE examination processes for which reconstruction is unnecessary or not revealing.

> **Reconstruction is unnecessary when the existence or number of items of specific content is the issue.**

For example, when asking whether or not contraband is present, search alone is adequate to the determination, and reconstruction is unlikely to be revealing. However, when the question is of whether the suspect could have had use of or access to the contraband present, reconstruction may be revealing in determining whether the environment allows such access or use.

When is reconstruction needed or revealing?

Similarly, there are cases when reconstruction is clearly the approach of choice.

> **Reconstruction is most revealing when a sequence of events is involved.**

If there is a question as to whether the O-traces reflect specific sequence(s) of events, even though people who have expertise, experience, skills, and knowledge of the issues at hand may believe that the O-traces are consistent with some set of events, a reconstruction may be revealing in terms of confirming or refuting the expressed opinions of those people and the actual consistency or inconsistency of the O-traces with C-traces.

Multiple runs of reconstructions may be additionally revealing when there may be many possible sequences and exploring the space of those sequences may be revealing, or when the nature of the

events in question are statistical, or in other words, the causal relationships are not precisely known.

As a fundamental issue in reconstruction, it is important to know and be aware of the fact that:

Software often does not do what people think it does

This includes the designers, implementers, and experts on that software. Reconstruction helps to resolve the difference between speculation and actual behavior. Even the seemingly obvious interpretations of O-traces often turn out to be inconsistent with the results found in reconstruction. In interpretation, reconstruction can be particularly helpful in increasing certainty.

The class approach and assumptions

An approach to reconstruction that may help to facilitate an effective process, is to use knowledge of how systems in general, and specific parts of systems of particular interest to the matter at hand, do things in most cases. Based on this knowledge, the examiner can more efficiently go through event sequences that are likely to produce C-traces consistent with the O-traces of interest.

The general hypothesis of the class assumption is of the form:

(1) A given sequence of events produces a member of a given class of C-traces

(2) The O-trace of interest is also a member of that class

<u>This does not imply causality!</u>

Just because the O-trace is contained within the C-trace envelope produced by the reconstruction, does not mean that the reconstruction is the only way that the class of C-traces or the specific O-trace can be produced. It only means that it is one way that the O-trace may be produced. Further, unless the actual O-trace is produced, it doesn't even mean that the O-trace can be produced. It only means that the members of the class containing the O-trace can be produced.

Reconstruction can help to determine what could have happened and what could not have happened based on an available O-trace. To the extent that the reconstruction is accurate, it can potentially

rule out certain event sequences by refuting the consistency of an O-trace with the envelope of C-traces for the event sequences reconstructed. To the extent that the examiner can credibly extend the specific C-traces and event sequences to larger envelopes, that are consistent or inconsistent with the O-trace, larger classes of hypotheses may be confirmed or refuted.

There is an underlying question to be addressed with regard to these class approaches in that:

> **The use of classes of events and traces assumes properties of the digital world that are not consistent with the general things we know of information physics**

These assumptions about classes call for assumptions about information physics that may only be reasonably held to be true under limited circumstances and with assumptions that should be stated as such and either confirmed to a reasonable extent or, at a minimum, recognized as possibly being wrong. This should be stated in reports of examination, and characterized as to the nature and extent of errors likely to result.

Assumptions about properties typically made

Information physics assumptions commonly made in reporting on class results with regard to reconstructions include, without limit, those shown in Table 8.2. Other areas are not included in this table.

Digital World	Reconstruction assumptions
Finite time granularity (the clock)	Time granularity at finer grain than the clock does not matter
Finite space granularity (the bit)	DFE is perfect at the granularity of the bit
Exact copies, original intact	Duplication may occur at any time, but is ignored as irrelevant unless indicated by the O-traces
Theft without direct loss	Theft can happen at any time but is ignored as irrelevant unless indicated by the O-traces

Digital World	Reconstruction assumptions
Finite (fast) rate of movement	Unless rates are specifically relevant, whatever time passes is treated as ideal
An artifice created by people	The artifice is exactly as the one reconstructed
Finite State Machines (FSMs)	The real FSMs operates exactly as the ones reconstructed
Homing sequences may exist	If homing sequences occur, they occurred in the original
Forward time perfect prediction	Forward time perfect prediction within the defined class
Digital space converges in time	Digital space proceeds as the reconstruction
Results are always "Exact"	C-traces may vary over an envelope of futures
Time is a partial ordering	Time is any of a class of partial orderings within an envelope
Errors accumulate	Errors accumulate within the class envelope of futures
Representation limits accuracy	Representation is as in the original
Precision may exceed accuracy	Precision is to within a class of future envelopes
Forgery can be perfect	Reconstruction is what really happened, to within the limits of the envelope
DFE is trace but not transfer	DFE is C-trace
DFE is circumstantial	The reconstruction is within an envelope of the circumstance
DFE is hearsay	Reconstruction is not hearsay

Digital World	Reconstruction assumptions
DFE cannot place a person at a place at a time	Reconstruction assumes what took place
DFE can show consistency or inconsistency only	Consistency within an envelope is the question at hand
Probability is dubious	Probability leads to and allows definition of the envelope
Content perfectly compressible	Anything compressing into the same result is equivalent
Digital signatures, fingerprints, etc. generated from content	No assumptions outside of those associated with these systems
Context tends to be global and dramatically changes meaning	Context producing consistent class traces is in the original context class
FSMs come to a conclusion	Same class FSM produces the same class C-traces
Hardware fault models from computer engineering	Only modeled faults exist
Time and space tradeoffs known	Time and space tradeoffs are reflected in C-traces if timing is considered
Near perfect virtualization and simulation possible	Only modeled virtualization
Many nearly or equivalent FSMs	Only modeled FSMs
Undecidable problems	Everything decided by execution of FSMs
Computational complexity limits computations	Forward computations execute as the originals did
Everything is decidable	Same class decisions produce same class C-traces

Digital World	Reconstruction assumptions
Consistency is guaranteed	Consistency of C-traces to O-traces implies same class inputs, states, outputs
Completeness is guaranteed	Constructed runs are complete and adequate to the purpose
Time limits on achievable results	The number of runs defines the time of the reconstruction
Fault tolerance by design	No untested hardware faults
Accidental assumption violations	Untested assumptions are not relevant
Intentional assumption violations	Untested intentional violations do not exist
Discontinuous space	Discontinuities not tested remain within the defined envelopes
Discontinuous time	Discontinuities not tested remain within the defined envelopes
Minor differences amplified near discontinuities	Differences not tested remain within the defined envelopes
Major differences suppressed away from discontinuities	Differences not tested remain within the defined envelopes
Identical use of an interface may produce different results	Differences not tested remain within the defined envelopes
Ordering may be reversed	Ordering not tested remains within the defined envelopes
Value sorts may be reversed	Value sorts remain consistent
Actuate-sensors loop errors	Errors not tested do not exist
Sensors/ actuators limited in physical properties	Untested differences remain within defined envelopes

Table 8.2 - Common information physics assumptions in reconstructions

It seems clear that there are many common assumptions made in reconstruction and that these represent potential weaknesses

associated with the reconstruction process and results. Nevertheless, reconstruction is the only method available to the examiner for testing hypotheses experimentally, so it becomes the duty of the examiner to understand and deal with these issues and assumptions in their reconstructions.

Key properties in reconstruction

Many of the assumptions with regard to reconstruction may be addressed by bringing clarity to a smaller set of key properties. The description used earlier of failed login attempt audit trail traces is a starting point for characterizing these issues.

Identify a test that will confirm or refute a testable hypothesis

The purpose of reconstruction is to construct an experiment that tests a hypothesis. Fundamental to that approach is the notion that the results of the test will refute the hypothesis if they disagree with predicted results given by that hypothesis. Thus the examiner should first:

- Identify the hypothesis being tested.
- Use the hypothesis to predict the production of traces and the non-production of other traces.
- Identify a test that will produce predicted traces if the hypothesis is true and produce other traces if the hypothesis is false.

Bound the test

In order to implement a reconstruction that will produce useful results, the tests must be bounded in a variety of ways. Common things that should be bounded include, without limit:

- The test should be bounded in time and space.
- The accuracy of the measurements required to confirm or refute the hypothesis should be determined.
- The granularity of time, space, and measurements should be determined so as to meet the accuracy requirements.
- The deviations from expected results that produce confirmations and refutations should be identified.

- The envelopes of inputs, states, outputs, and traces that confirm or refute the hypothesis should be explicitly stated.
- The number of tests and other related issues required to meet testing conditions, such as to resolve uncertainty, should be identified.

Construct a test environment

A test environment must be constructed for a reconstruction to be performed. Depending on the nature of the test and related conditions, this can range from simply running a program on a computer to creating a physical network and building up sets of computing devices. The key thing is to make certain that the environment created for the reconstruction does not interfere with the utility of the test and that it meets the other requirements of the identified bounds on the test. Some of the things that drive up complexity in this stage include, without limit,

- How many systems of what sort are needed?
- How many different variations are required?
- What are the configurations and at what level of detail are they to be configured?
- How will repetitions be undertaken and to what level of fidelity in what ways?
- How will inputs be generated?
- How will traces be recorded?
- What has to be stored and where will it be stored?
- Is interaction with outside systems, like the Internet or places on the Internet required, and if so, what properties of it are required and how will they be tracked and assured?

Again reverting to the example discussed above, if logins on the particular system operate through a network authentication mechanism, like a remote access server (RAS) that uses an authentication server (e.g., a AAA server) and an identity management (IdM) system, then depending on what essential properties are required, if any, from these systems, the examiner may have to configure a set of systems, provide them with

configurations, and implement a network complete with licensing servers and authorized copies of expensive software. Or alternatively, an existing server configuration from the actual environment might be usable. But each of these has its problems in terms of accuracy, potential changes since the time of the O-traces, software patches that were or were not in place at different times in different systems, and the list goes on.

Each of these things might impact the generation of audit trails associated with the C-traces, and as such, they represent a large class of C-traces that likely cannot actually support the testable hypothesis at issue. That is, even if the test environment was created, with all of these uncertainties, what certainty would the examiner have that a result showing that traces are present for 2 failed login attempts means that the lack of traces indicate that 2 login attempts were not present in the system that generated the O-trace? And if no trace occurred with 2 failed attempts but a trace appears in 90 seconds for 3 failed attempts, would this confirm the hypothesis even though the hypothesis was limited to 60 seconds? Of course it cannot confirm the hypothesis and must in fact refute it, or the process is not scientific.

In terms of schedule, the importance of this particular test to the matter at hand must drive the process. The examiner identifying that this is the key to the case, in discussions with legal counsel, must identify what can be reconstructed and to what level of certainty the result needs to be known before going down the expensive and time consuming route proposed here. For example, the examiner might choose to simply look in the system's administration manual and read about configuration issues with the particular environment to identify that: "... According to the manuals and configuration guide", which the examiner should then cite, "failed login attempts are logged when ...".

This thought process must be considered in some manner by the examiner in creating the test environment. Alternatives should also be considered, including not resolving this particular issue, changing the hypothesis to something that is more readily tested within the schedule, or seeking to get a stipulation from the other side with respect to the underlying legal issue.

Virtualization is also often a good choice for reconstruction, particularly when performance or specific hardware is not a key issue in the case. By using virtual computers within a test environment, even substantial networks can be generated with relative ease and without the need for additional hardware or software. Many virtual environments may be created by the examiner, and substantial collections of such environments may be generated over time and be reused with only minor changes. This makes reconstruction very efficient for many standard or previously examined situations. Virtual environments are also very handy for dealing with a multiplicity of different patches applied in different sequences over time, and for bringing configurations and software into and out of environments that do not support general purpose networking. Emulators for many hardware and software systems are freely available, and others may be purchased for low cost.

There are also problems with virtualization in reconstruction in that virtualized environments may be substantially different from real environments, they typically aren't as well tested as the original systems they are based on, and they introduce an additional layer of software and potential uncertainty into the process. Again, the key is whether and to what extent the virtual environment gets at the issues in the case without introducing so much uncertainty that the results are inconclusive, or perhaps even misleading.

Perform the tests

The tests must then be performed to get results. It is usually desirable to have reproducible results, and in this case, forensically sound images of the configuration prior to testing may be made in order to allow the configuration to be more rapidly regenerated. Once an environment has been properly constructed and input generation understood and prepared, the tests may be run and the resulting C-traces generated. The C-traces must then be collected and properly controlled and preserved for analysis.

Analyze the C-traces against the hypothesized C-traces

The constructed traces are analyzed using the same sort of analysis methods discussed in the Analysis Chapter to compare them to the hypothesized C-trace envelopes predicted by the hypotheses. Depending on the metrics selected for consistency

checking, the analysis results indicate that the C-traces are consistent or inconsistent with the envelope or that the test was inconclusive. Regardless of the outcome of the test, conclusive results should be provided to counsel and, depending on the legal situation, may or may not be reported to others.

There are instances in which inconclusive results, even if reported, will not be particularly probative. An example of a meaningless inconclusive result is a result indicating that the test could not be completed because of a power failure. Similarly, if a test fails to achieve adequate accuracy to draw a conclusion, or if it fails because of a configuration error, incorrect input syntax, the inability of the examiner to provide data at a sufficient rate, or for other technical reason not having anything to do with outcomes, the test is normally meaningless in terms of getting at the question at hand. If, however, the reason for the failures of the test are revealing with regard to the matter at hand or the reliability of the process, methods, or procedures, the results are presumably reportable.

It is quite common for examiners to make simple mistakes, such as typographic errors, and produce test results that are irrelevant. For example, suppose that the examiner typed in a different user identity on the third login attempt within 60 seconds in the example reconstruction. While the test might show that login attempts with two failed attempts within 60 seconds do not generate relevant traces, it does not show what it was intended to show with regard to the third login attempt. It is simply unrevealing. People are imperfect, and that's just the nature of the being.

Optionally loop

After reconstruction is done and tests performed, or at other steps along the way, there may be good reason to loop back and revisit earlier stages.

For example, in examining the results of a reconstruction of the login example, it might be determined that in one case the third login failure was reported, and in another case it was not. Upon further analysis and interpretation, the examiner may have a hypothesis as to why the recording did not take place in one instance and did in the other. This then could be tested in an additional reconstruction, or perhaps it might merely require an

additional test run of the current reconstruction. Perhaps the traces are only generated when the attempts are made over the network. Or perhaps when they are made over different channels, such as via the file transfer protocol, via telnet, and via the console, the traces are not generated. Perhaps when made from the same interface, they are generated. These hypotheses, to the extent that they are relevant to the matter at hand and compatible with the schedule, may be generated and tested quickly.

Uncertainty in reconstruction

Clearly, there is great potential for uncertainty in reconstruction. The hypotheses identified for failed login logging may have many more possibilities, and each may produce very different outcomes. But a few things help to dramatically reduce the uncertainty and the number of different hypotheses tested.

Perhaps the most important issue to be addressed is that the reconstruction reflect, as closely as necessary, the hypothesized events. If the case has to do with console logins, network logins do not have to be reconstructed and should not be, at least not as part of the reconstruction related to the issue identified. If all login attempts of interest are using the file transfer protocol, other protocols need not be tested.

There are also limits to how many test runs can be completed within the available schedule. More test runs normally increase the accuracy and certainty of the results, so in that sense, more is better. To the extent that repetition refutes what was previously confirmed, it is highly informative and, subject to verification of experimental method, should be used to produce statistical data with regard to the methods in use.

Coverage of the input and state space of the envelope is also an important thing to consider because, with low coverage, the test has more limited applicability. Coverage is generally defined as the number of sequences tested divided by the total number of possible sequences that can be tested, and is based on a fault model that defines the classes of sequences to be differentiated. For example, even if testing was done with one user identity and an incorrect password, this may not work for all user identities or all incorrect passwords. The user identity "supervisor", or "root", or

"administrator", for example, might have special rules so that, even if the disk has no space left to record failed login attempts, these users will still be able to login from the console in order to recover the system to normal operation. If the fault model includes this issue, then the coverage will be based on including those tests, while if the model does not include them, coverage results will be different.

The notion of coverage then depends on the model of the envelope used to determine what to test. If the model of the envelope is inaccurate with respect to the issues in the case, the results may also be inaccurate. Depending on the specifics of the model, some of the tests may be more important because they cover a larger portion of the space or are otherwise modeled as more important.

Because reconstructing all possible sequences of events is not usually feasible, a representative sampling will almost always be the limit for testing. The examiner may identify what constitutes a representative sample based on the model, but again, if the model is problematic, the results of the tests may also be problematic.

One approach to limited meaningful reconstruction

The discussion to this point has seemingly showed just how difficult it is to carry out a meaningful reconstruction within reasonable limits of schedule. At the same time, there are many cases when a small and meaningful reconstruction can be done to show a small point of interest to a case with far less time and effort.

For example, suppose an issue in a case is the potential for a log record to be recorded when 3 failed login attempts are made, that the system of interest is known to use a particular operating system and version with only local authentication, and that the system was forensically imaged in collecting evidence for the case. In such a case, a simple test can be done to duplicate the forensic image of the system, install it in similar hardware or in a virtual environment, start the operating system, and try to login 3 times with an invalid user identity and password. After this test, the system logs are examined for traces of the failed login attempts. The result is then presented something like this:

> I received what I call Item D, a file asserted by [THEM] to be a disk image [describe everything necessary including the tests performed]. Upon examination of the resulting image, I found [either "no" or "the following"] traces of the failed login attempts. [If traces were found, list the traces found here].
>
> This results of this reconstruction are [either consistent or inconsistent] with [identify the relevant event(s)]. Subject to the limits of a single test of this sort, this appears to indicate that [the relevant conclusion]

This approach is simple and clean, and if reasonably well done, it properly characterizes what was and was not done and the results. It is also important to make certain to identify the limits of such results for clarity. In the section of the report discussing the reliability of the techniques applied, it is reasonable and prudent to indicate that a single test or demonstration provides only limited insight into the possibilities of what could have taken place. And of course, if it took 25 experiments to get the identified result while the remaining experiments all showed different results, it would not be appropriate to provide such a misleading result in the first place.

A slightly more complex reconstruction

An example of a slightly more complex but practical reconstruction is to show that something is possible. For this sort of case, a single instance of an execution that produces a confirmation is adequate to show the point. For example, suppose a legal matter depends on the reliability of some trace because that trace links a suspect to a crime. A single refutation of the reliability of the trace would make it far less probative from a standpoint of the issue at hand.

A reconstruction in this case would be adequate if it simply showed that, under some circumstances that are not completely bizarre, the trace was not consistent with the hypothesis about it. On the other hand, a small number of consistent traces under limited circumstances would not prove that it is reliable. In such a case, the report of a single experiment that shows that the trace is produced when the suspect is not involved becomes a very big deal. It might be explained something like this:

> I received what I call Item D, a file asserted by the State to be an exact copy of a file written by Defendant. The basis for this claim is the presence of Defendant's user identity in an internal structure within the file.
>
> To determine whether the presence of such an identity in such a location in such a file so indicates, I edited a copy of Item D by adding several paragraphs, deleting several other paragraphs, and saved the resulting file, all using the same software identified by the State as that used by Defendant. Upon examination of the resulting file, I found no traces of my activities at the identified location within the file, and the trace of Defendant's user identity remained.
>
> I then performed another experiment in which I took another file of the same type, also asserted as written by Defendant by the State, modified that file using the same software identified above, and produced a document which, when printed out, appears to be identical to the one the State claims shows that Defendant performed these acts, and saved that file. Upon examination of the resulting file, I found no traces of my activities at the identified location within the file, and the trace of the Defendant's user identity in the identified location remained.
>
> This results of these reconstructions are inconsistent with the State's claim that the presence of the identified User identity in the identified place within the file indicates that Defendant wrote the contents of the file, and consistent with another party writing the file in question.

It is noteworthy that very little effort is required for such an experiment and that the presentation of this result seems quite compelling. It is now incumbent on the State to overcome this refutation of their claims, and this may be very hard to do. In essence, it shifts the burden of proof to the other side.

It is also important to note that nothing stated or demonstrated in this example indicates that Defendant did not in fact do what the State claims. Indeed, these results are also consistent with Defendant writing the file in its entirety and nobody else ever writing the file at all. This is something that the State might want to bring up

in such a case, and of course the defense would counter it with the burden of proof and what the State indicated was the meaning of the traces, and so forth. The State could also bring up all the inadequacies of the experimental process used in the reconstruction, but this would likely change little or nothing about the result. After all, what experiment did the State undertake to prove their case?

A reconstruction to determine how to reconstruct

In many cases, the available traces are inadequate to directly do an accurate reconstruction to the level of granularity desired. For example, a document may be provided in digital form, but the version of the document editor and system used to create and manipulate it may not be. Because document editors are updated periodically and may work differently in different operating environments, to the extent that traces from within the document are key to the legal matter, reconstruction with precise versions may be probative to the matter at hand. In such cases, traces from the document may be used to identify the particulars of the operating environment to some level of accuracy and certainty.

Such cases normally start with bounds on the environment. For example, events may indicate that the operating environment was a particular operating system, the date and time information from events may limit the possible versions based on availability at that time, and the networks to which the system was attached may indicate minimum version information.

Starting from that point, the set of candidate environments may be tightened further by reconstruction. A typical process is to start with the earliest possible version, gather updates and patches from historical archives, apply one after another update or patch, and test with each of these by reproducing the activities that events indicate generated the document. After each such test, C-traces are compared to O-traces to identify characteristics that are consistent or inconsistent with the operating environment, and the result may be a narrowing of candidates. As an example of the sorts of traces that such a process might use, many operating environments record information about the environment, such as indicators of interfaces used, library details, or even date and time stamps with

different initial offsets. A specific example would be the storage of the MAC address of a system Ethernet card in a document file, which limits the possible dates of the activity that produced that identifier in the document to the earliest date at which the particular Ethernet card was available in the market. Other events and traces may narrow things further, for example, by documenting the actual purchase of the Ethernet card.

By identifying things that change within the document when the same test processes are used for different versions of the operating environment, candidate indicators are identified. These are then compared to the O-traces to see whether these indicators are present or absent. Differences in C-traces are then associated with particular operating environment versions and similar O-traces sought to make an estimate of the version and patches of software used. Once the candidates have been reduced to a small enough number, reconstruction can proceed to test the issues in the case against the O-traces with a smaller number of versions that need to be reconstructed. The earliest time for identified activities producing traces is also bound by version and patch availability.

Legal restrictions and reconstruction

For various reasons, there are limits on what can be done to be thorough in reconstruction.

The DMCA and other laws

For example, an excellent way to understand and deal with the issues of coverage would be to examine the inner workings of the FSMs involved and determine their branching characteristics to determine how to create a test set with defined coverage. Unfortunately, this is against the law in the United States except for law enforcement and intelligence agencies, as restricted by the Digital Millennium Copyright Act. There are also legal restrictions from copyright law that potentially limit the ability to use the same software licensed for use in one system within another system, and there may be contractual liabilities associated with reconstruction when running software or examining content that is restricted for one reason or another.

Network reconstructions and access restrictions

In networked environment that are quite common today, reconstructions involving the execution of mechanisms that cross network boundaries are problematic for several reasons. These include, without limit:

- The networked environment may change with time, and thus experiments may not be repeatable or reflective of the time frames at issue to the matter at hand.
- Content stored in networked locations may change with time, and thus the validity of results that depend on input from the network may not be valid.
- Networked operations may include access to remote systems to which access is not legitimately available for the purposes of the matter at hand.
- Networked operations may expose other systems to harms caused by the reconstruction, such as the introduction of malicious code, the introduction, alteration, or destruction of undesired records, and consumption of resources.

Inadmissible evidence with admissible reconstruction results

Another interesting legal issue is that, even in cases where the evidence is not admissible for one reason or another, the results of a reconstruction may still be admitted as part of an expert opinion.

Challenges to reconstructions

In challenging the use of a reconstruction, there are many approaches that may be undertaken. As any other scientific evidence, the results of a reconstruction must meet the standards of admissibility, and this includes the provision of information about possible sources and types of errors. Some of the questions that are likely to be asked include, without limit:

- Is it just fantasy or a simulation?
- Is it more probative than prejudicial?
- How many possibilities are there and how many tests did you do to cover all of those possibilities?

- Is the result displayed reflective of reality? All realities or just one of them?
- Is this evidence or entertainment?

The answer to these and other related questions may have to be addressed by the examiner using reconstruction, and the examiner should be prepared to explain the issues in a sensible fashion that will pass the legal tests of admissibility, both of the evidence and the related testimony. The examiner should expect to be questioned at least to the extent that challenges are identified. As the field progresses, it is likely that other challenges including, without limit, all of the assumptions made that vary from information physics, and all of the cognitive error types, may be used.

What does a DFE reconstruction laboratory look like?

A laboratory to support DFE reconstruction is typically quite different from a laboratory used for other aspects of examination. For analysis, interpretation, and attribution, the examiner typically uses an environment most suited to the tools that make their tasks easier, faster, more reliable, and on which they have been trained. But for reconstruction, the laboratory has to provide environments that are sufficiently similar to the environments in which the O-traces were generated to provide probative C-traces. This typically means that the reconstruction laboratory will include:

- Many devices of different ages and types.
- Many device emulators within larger physical machines.
- Many standard configurations forensically imaged on servers and loadable onto devices or emulators to support rapid creation of test environments and repetition of tests.
- Collections of startup media, installation media, patches, and manuals from over a long time frame.
- Mechanisms designed to allow for automated recording, playing, and replay of input sequences.
- Mechanisms for recording activities at a high level of detail within systems and networks.

Computer museums, recycling facilities, and swap meets are three places to look for the things that are useful in such a laboratory, but today, there are few if any facilities that really support reconstruction in a meaningful way.

Recent results from funded research have produced university-operated environments in which large pools of virtual machines and capabilities are available for educational use, and this notion has been proposed to follow along the lines of national-level supercomputing capability in which large-scale resources for forensic experiments and reconstructions are made available to authorized users.[340]

There are certainly challenges associated with the use of such facilities for reconstructions, but on the other hand, the cost of maintaining large-scale capabilities for reconstruction are limited to organizations with the resources to create and operate such facilities over time. As such, shared environments may be the only practical solution. Chain of custody and related issues appear to be readily managed in such environments, but issues of repeatability and reliability of results, control over tools, and related matters have not been well addressed in such facilities to date.

What we can and cannot reasonably say

With all of the issues, limitations, and complexities associated with reconstruction, the examiner who uses these techniques must be able to make statements regarding those experiments that are meaningful to the legal process without those statements collapsing under the weight of all the doubt associated with the experimental process.

Of course the examiner who never uses reconstruction has no experimental basis for their results, and as scientific statements, they are subject to questions about whether this examiner actually knows that the things they are asserting are true. While statements like "I searched Item X for string Y using tool Z and did not find string Y within item X." may be extremely useful, interpretation is far

340 K. Nance, B. Hay, R. Dodge, J. Wrubel, S. Burd, and A. Seazzu, "Replicating and Sharing Computer Security Laboratory Environments", Proceedings of the 42nd Hawaii International Conference on System Sciences - 2009, see also The "Assert" Lab - details at: http://assert.uaf.edu/

harder to present than analysis without an experimental basis. And a single refutation by experiment of an interpretation made without experimental basis is potentially devastating.

Reconciling the need for experiments with the difficulty of all the assumptions is where the language used, the statements used, and the manner in which things are done come into play.

Who did what and how

It is always appropriate to indicate, in as clear and concise language as possible, the facts associated with the activities performed, and the source of the O-traces. For example:

> I received a file named "JJ.jar", referred to herein as Item X, which contains what appears to be a program and associated data designed to be interpreted in the "Java Virtual Machine" (JVM) interpretation mechanism.
>
> According to the SECOND AMENDED COMPLAINT (Item A, page 18, line 23), "JJ.jar was used to download files from the Defendant's computer..."
>
> I executed JJ.jar in [test environment], which is consistent with the situation identified in Item A, with the Java Console enabled to track the execution of the Java program, to determine whether it attempts to download files or otherwise execute system calls compatible with such an activity.

This sort of report or statement is intended to be of factual nature, sufficient for others to attempt to repeat the experiment for themselves. It need not be at a level of detail that is excessive. To the extent that many details are available and helpful in understanding what was done and repeating the reconstruction, they should be included in an appendix or a more detailed portion of the report. In this way, experts can review and repeat the reconstruction, but the judge and other triers of fact don't need to wade through excessive detail to get to the point.

The results of the experiments

Once the background information is identified, results are typically presented in a simple form. Here is a continuation of the example:

> In execution, JJ.jar did not undertake any activity or make any system call that in any way indicated an attempt to access any computer network whatsoever.

Of course this is a very positive outcome for the defense in such a case, but it has some obvious limitations associated with inputs provided and other similar factors. To the extent that multiple inputs were tested that are consistent with the claims made, this is helpful in increasing coverage, but again, these may be placed in the appendix with a simple statement about testing under a variety of conditions like those identified by the other side.

The implication of these results as interpretation

This is the area where examiners sometimes go a bridge too far by making claims not justified by the results. Here is a reasonable claim given the results identified above:

> The execution of JJ.jar in the reconstructed environment was inconsistent with [the other side's] assertion that "JJ.jar was used to download files ..."

To the extent that more definitive statements can be made, they would have to be justified by something beyond what is included here, and of course this statement is hardly definitive in that it does not say that JJ.jar did not or could not download the files. It only states the inconsistency between the results of the reconstruction and the assertions.

Identifying assumptions and limitations

In reports regarding reconstructions, and in other scientific reports, there should always be information about limits and assumptions. Here is the sort of thing that might be stated about reconstruction:

> From a standpoint of identifying possible sources of error and reliability of these tools in this context, I have found that [details related to other tools might go here...] Reconstructions are also subject to various assumptions, including without limit, specific assumptions and accuracy figures identified for the reconstructions performed herein, and common assumptions elsewhere in the literature.[341]

341 F. Cohen, "Digital Forensic Evidence Examination - 2nd edition", ASP Press, 2010, ISBN 1-878109-45-6] (the last edition of this book).

The report should appropriately identify sources of errors of all sorts. But rather than detail all possibilities, it is reasonable to cite external sources of assumptions and errors and to justify the presence of language about such errors based on other reports and publications. A key element in understanding this issue is that, by adding this information to reports, the other side is also forced to provide information on sources of error in their reports. To the extent that such sources of errors are identified in examination reports, the failure to so identify them in their reports indicates a lack of scientific rigor and possibly leads to exclusion.

Of course it is to be expected that, in challenging these results, questions about sources of error will be asked, and it is incumbent upon the examiner to be able to identify, without becoming defensive, that all science is imperfect, and that to make any such statement without recognizing that there is potential for error would be unscientific.

Questions

1. As the experimental branch of DFE examination, what role does reconstruction play in advancing the science of digital forensics?
2. Given the problems with reverse time reconstruction, how might reverse time still be used? Are there cases where a small number of steps can be used or when the input classes are similar enough to be used regardless of the obvious problems?
3. Is the approach of Table 8.1 a sensible way to go about understanding the limits of a forensic technique, or can no technique ultimately hold up to this level of scrutiny?
4. How can an experiment be unrevealing?
5. Given that an O-trace should never actually be the original evidence, but only an exact bit-for-bit image of it, is this a problem for reconstruction?
6. Since a precise initial state is almost never known, what state assumptions must be made in generating an C-trace and how do those assumptions get disentangled from the difference between the C-trace and the O-trace to identify the difference between consistency and inconsistency?

7. How important is time bounding in reconstruction? What if a test had no time bounding and it was simply run until a C-trace consistent with an O-trace was found? Wouldn't this be just as good as a time bounded reconstruction?
8. Since the number of possible initial states is so large, how can any experiment be repeated without an identical initial state?
9. Why is it that reconstruction is revealing when a time sequence of events is involved and not when no time sequence is required?
10. How does the class approach resolve problems with reconstruction and what new problems does it introduce?
11. If class consistency does not imply causality, how can attribution be shown with reconstruction?
12. Using the assumptions in Table 8.2, under what conditions can a reconstruction be done so as to eliminate each of these assumptions?
13. Is there a systematic way to identify a proper test for a situation, or is this simply a matter of the training, knowledge, experience, expertise, and education of the examiner? If the latter, how can this be used as the basis for judging the value of the experiment in getting at the issues in the case?
14. A test that is not bounded in some way potentially results in a logical fallacy or other similar problem. Identify the specific reasons for each of the bounding conditions identified in the text. Are there other bounds that are required? If so, what are they and why? If not, how do you show them sufficient?
15. Constructing a test environment seems like a lot of effort. Is such effort justified in every case? In which cases is this likely to be used, and how much time and effort is likely to be spent in reconstruction? If time is not spent in reconstruction, how can the assumptions of other parts of examination be confirmed or refuted?
16. In performing the tests, a simple mistake can lead to a completely wrong answer. How do we make certain that the test is properly performed?
17. Is the comparison of C-traces against hypothesized C-traces always easy to do? What is involved?

18. Suppose the comparison method is not defined prior to testing. Is there a danger that the creation of the basis for comparison after the test will skew results toward the examiner's viewpoint? How can this be avoided?
19. If reconstruction is the experimental part of the science of DFE examination and there are unquantified uncertainties involved in it, how can this be called a science at all? How then do we quantify the uncertainty?
20. The example reconstruction results seemed pretty simple. In fact, they seemed too simple. What was missing and how does this invalidate the results presented? Or does it?
21. How likely is it that reconstructions will be thrown out because they are fantasies? What are the key parameters associated with reconstruction that make it valid scientific evidence, and when these are missing, to what extent is it no longer scientific or admissible?
22. Given the wording used in the sample statements, do any of them go a bridge too far? Analyze these statements against the common fallacies to identify possible weaknesses.
23. Given the weaknesses identified in the last question, how can the examiner using these statements still make them credibly? What supporting information is required? What will rehabilitate them given the devastating counters provided in the answer to the last question? And how will these responses be countered?

9 Tools and process

Digital forensic evidence examination uses tools, and those tools have inherent limitations that the examiner must understand and deal with in order to be effective at providing the legal system with accurate information.

Ultimately, tools used in legal work are subject to review for the methodology underlying their use, their history, pedigree, and reliability, the manner in which they are tested and calibrated, and their function and limitations, specifically including error rates.

The National Research Council recommends that:[342]

> "As a general matter, laboratory reports generated as the result of a scientific analysis should be complete and thorough. They should contain, at minimum, “methods and materials,” “procedures,” “results,” “conclusions,” and, as appropriate, sources and magnitudes of uncertainty in the procedures and conclusions (e.g., levels of confidence)."

How then do we accomplish this?

Unlike evidence collection methods, which have been largely characterized, most of DFE examination today is not standardized in any way. Each case today has the potential to use slightly different tools, methods, and techniques.

This is particularly true on the defender's side of legal matters, because the prosecution (or plaintiff in civil matters) does not have the burden of proving that all traces are consistent, that all traces are consistent with all events, that all events are consistent, or even that reconstructions are correct. Rather, they only need to do enough to make a case seem clear to a potential jury and to get past the admissibility requirements of the court.

To many, it may seem that the burden of proof has, for these reasons, shifted to the defendant, and that it is far easier to make, or make up, a case in digital forensics, than to defend against one.

342 Committee on Identifying the Needs of the Forensic Sciences Community, "Strengthening Forensic Science in the United States: A Path Forward", ISBN: 978-0-309-13130-8, 254 pages, (2009).; Committee on Applied and Theoretical Statistics, National Research Council.

Today, this is largely true, but most examinations to date have gone unchallenged. This seems likely to change very soon.

Clarifying the limitations of examination

In the case of DFE, following the NRC recommendations means clearly identifying the methods, systems, and software used in analysis, the procedures, their results, conclusions based on those results, the known and potential sources of errors in analysis, how far off the results could potentially be based on these sources of errors, and how certain it is that any conclusions are right.

Unfortunately, there is no widely accepted systematic method of doing this for DFE examination methods today. We can certainly say, for example, and as appropriate:

- The methods described in this book and other papers have been peer reviewed, published, and carefully undertaken.
- The operating system and software used in the analysis is widely used by millions of people and companies each day and has proved reliable for the sorts of functions used in this case.
- The examiner has found all of these to be reliable for the purposes used and that these systems have been used and tested over a period of years to verify that they operate as identified.
- The methods were repeated with identical results using redundant mechanisms and approaches (which should be identified where reasonable) and yielded the same or nearly the same results (explain any discrepancies).
- I checked these results and believe them to be true and correct.

All of this, of course, assumes that it is true of the specifics in the case under consideration. But presumably, it is also appropriate, and perhaps necessary, to add something to the effect that:

- Computer programs sometimes produce results that are off by one or otherwise different than what might be attained by hand counts, either because of programming errors that are

not detected even after a long period of use, or because of differences in interpretation of what constitutes things like "words", "lines", and so forth.

- Computer hardware, operating systems, libraries, software, and other components are imperfect, and even though they sometimes fail in modes that are catastrophic, they may also fail in ways that produce incorrect answers without other indications.
- Limitations and intended uses of the tools applied are often not known to the examiner, including without limit, the maximum sizes and numbers of things they can handle properly, the formats they work on, and the mechanisms they apply.
- User errors, including without limit, typing errors, incorrect or otherwise imperfect specifications or syntax, incorrect combinations of output from one tool used as input to the next tool, failure to provide surrounding process to assure that results of different tool uses don't improperly overlap, and other similar errors, may produce results that are inaccurate.
- It is impossible to perform every test that can be conceived of with respect to traces because of the large number of different possible ways in which actions within computer systems may take place.
- Any of the particular results of information physics that may introduce errors into the results that specifically apply in this case can also be listed.

Here is an example:

> In carrying out all of these examination, I used an [XX] computer running the [YY] operating system, the [list tools here], and the programs and mechanisms described herein. I have found all of these to be reliable for the purposes discussed herein and have used and tested these systems and tests over a period of years to verify that they operate as identified herein.

> From a standpoint of identifying possible sources of error and reliability of these tools, I have found that computer programs sometimes produce results that are off by one or otherwise different than what might be attained by hand counts, either because of programming errors that are not detected even after a long period of use, or because of differences in interpretation of what constitutes things like "words", "lines", and so forth.
>
> I have verified each of these results so as to reduce or eliminate such potential errors, and I believe that all of the results herein are accurate as stated.

Of course, if this is properly done, with DFE, there is every reason to believe that the other side can verify results by reproducing them, and that if their examiner finds errors in the results, they will be put forth as challenges to the extent that they are relevant.

The analysis discussion in this book has few examples of analytical errors with sound statistical characteristics that have been widely published, and the careful examiner should hand verify results of mechanisms that are known to be imperfect. For example, the symbol set and type identification issues are always potentially error prone except in cases where there is an event that asserts something of import and an inconsistency with it is found. Methods like the "JDLR" comparison of file extensions to content are problematic, and the elements of various toolkits used to search are subject to a wide variety of potential limitations, including searching across various kinds of file or block boundaries, the inability to test all possible orderings of parts of traces that may combine together, imperfect knowledge about the actual manner in which the particular computer system starts up, and the operating environment in use at any particular time.

Validation of examinations and examination systems

While the theories and methods presented here seem to be reasonable and reliable ways to identify consistencies and inconsistencies of internal (C) and external (D) types, in order to be reasonably applied in a legal setting, a further step is required.

Validity of consistency results relating to traces

The validity of the underlying statements and assumptions must be demonstrated for the particulars of the matter at hand. There are generally five common methods by which this may be done, ordered with the first preferred over the second over the third, and so forth:

1. Purely analytical results that directly show internal (C) or external (D) consistency or inconsistency are very helpful, simple to do, and very direct. They also tend to be very reliable and simple to confirm or refute. For example, an analytical result indicating the presence or absence of a string within a trace can be done very directly, and to the extent that this is at odds with an event, the interpretation is straight forward and hard to dispute. The method consists of identifying the event, the trace, the search tool and method, and the result, and drawing the conclusion that they are consistent or not. The same is true of counts of strings within traces and for orderings in time, in cases where there are consistent formats and time differentials are large enough so that most error modes do not apply. These results are also easily repeated with an independent tool and method, and doing this substantially reduces potential error modes. Limited hand review and sampling to verify these results also helps. So does sorting results and reviewing them visually to detect outliers.
2. Reconstruction may be done with the specific mechanisms and systems involved in the matter at hand to demonstrate that those systems produce results that are consistent or inconsistent with the traces found. This is preferred when time sequences or behaviors are at issue because, if the reconstruction is of adequate fidelity and properly undertaken, it will most accurately reproduce the specific situations and sorts of events associated with the traces. But this is time consuming and expensive and may require resources that are not readily available. It is at its best when simple tests refute claims, when it is used to bound time or other similar aspects of the matter, when it is applied to specific high-import issues when there is an opinion that the

tests support or refute, or when an event or trace seems inconsistent, but the inconsistency is based on an assumption that a reconstruction can clear up.

3. Statistical results using similar systems may be used to demonstrate, for example, that almost no messages within large collections in similar systems produce out of order time stamps in those message collections, with this result compared to the specific results found in the matter at hand. While this method may provide reasonable results, the nature of digital systems is inherently case specific and non-continuous. Therefore the common statistical assumptions of event independence, random stochastic processes, and normal distributions are not usually valid, and the degree to which they are approximations must be made clear with the mathematics of statistics adapted to the specific circumstances.
4. The examination of relevant hardware and software to make a determination based on that analysis about the consistency of the traces and events. This is problematic because of the complexity of undertaking this sort of effort in most modern systems and the legal implications regarding US Federal laws like the digital millennium copyright act (DMCA).[343]
5. The author or someone else who is familiar with the mechanisms can testify to establish the factual accuracy of the method based on their specific knowledge of how the mechanisms work. The problem with this is that even the authors of most current software may not know how that software works in all circumstances. If they did, then we would not have all of the software vulnerabilities we see in the media. In addition, the complexity of modern software in situ, the lack of compliance with specifications, the team approach commonly used for writing software, and the number of different specific circumstances that can be encountered, make it difficult for even the authors of

343 DMCA, H.R.2281, "Digital Millennium Copyright Act" (Enrolled as Agreed to or Passed by Both House and Senate), Available at: http://thomas.loc.gov/cgi-bin/query/z?c105:H.R.2281.ENR:

software to always be right about what it does under all circumstances. Those who are not authors but are merely experienced with the use of software and systems are even more problematic because they have even less specific knowledge.

Combining more than one of these makes the case stronger, particularly if they all give consistent results.

Validity of mechanism used to do the examination

Of course the tools used for the examination process have some of the same constraints in terms of demonstrating their validity for the purposes they are being applied to. While compiler compilers, search programs, and other similar components in widely used environments are suitable for the general class of things discussed, are widely used for similar purposes and relied on, and can be tested and used by the examiner to validate the results and calibrate them, the reliability of their application and the specific manner in which they are used may reasonably be questioned. The solution we advise for dealing with the issues involves any or all of the following:

1. Use tools that have been widely used and well tested, such as the Unix "grep", "awk", "wc", and shell programs, and other similar programs that are widely used and have been used for many years for similar purposes.
2. Test the tools on sample data of your own construction in advance of their use and confirm results after use to demonstrate calibration and validation.
3. After getting results, do redundant checks or other cross checks to verify that the results are accurate, such as; selecting samples from the results and making certain that they are correct by manual inspection, and doing counts with different counting mechanisms.
4. When reviewing a report or other deliverable, redo the key steps to verify that what is being written is accurate, and document that this was confirmed by the applied method.
5. Make sure the lawyer on the case examines your results at least on a sampled basis to confirm what you have provided.

6. Use an independent party or another member of your team who has not seen the matter before to confirm or challenge the results.
7. Create golden unit copies of the operating environments you use for digital forensics to reduce questions about what tools do, or do backups so that the system configurations and programs used can be provided for review by the other side.
8. Use other integrity techniques to verify and validate the tools, their operation, and the environment they operate within.

Combining more of these approaches will tend to produce greater acceptance because they confirm each others results with some level of independence.

If process is reasonably well explained, or if results are explained in a way that can be independently tested, there will be little doubt of their accuracy. For example, if a count of the number of lines containing a string is given as the results of a "grep" followed by a "wc" or a sort of the result, the other side can reproduce the results and confirm or refute easily. But if the result was also confirmed by examining the results with an editor that produces line counts and by visual inspection of the results, this redundancy makes the result harder to challenge. If a cross check with a completely different method is also used, it gets harder and harder to challenge. And if the other side finds a different result from the same process, the court can use a special master to resolve the issue, if necessary. In practice experts rarely disagree about such things, and once there is such a disagreement, the issue tends to be resolved rapidly.

Process controls

Ultimately, process controls are required in order for tools and techniques to be reasonably and demonstrably reliable. It is helpful for standard analytical processes to also be documented and repeatable. Even if some steps are not so well defined, the steps that are well defined can be documented as part of a procedure, and this provides increased assurance that the methods used are repeatable. If this is augmented with a checklist that is checked contemporaneously with any errors noted, this helps show

consistent use of process as well. A reasoned list of major process elements that is consistent with laboratory procedures used for collection and related processes that are fairly standardized includes, without limit:

- Use a defined and documented process
- Use tested components and tools
- Keep independent things separated
- Assure the purity of original and duplicated evidence
- Validate the purity of duplicated and derived evidence
- Use known test samples with known results
- Take contemporaneous notes
- Calibrate with known samples prior to use
- Clear results areas prior to use
- Use tools consistent with procedures
- Check results with redundant process after use

To get a sense of how widely this sort of process is accepted, the American Society of Crime Laboratory Directors / Laboratory Accreditation Board (ASCLD/LAB)[344] is one of the most widely respected accreditation organizations in the United States, and the methodology that they require is similar to this, except that it is oriented largely toward collection, preservation, and very limited analysis functions. Australia's NATA certifies electronic evidence laboratories with a similar very limited requirement.[345]

Defined and documented process

In order for a process to be considered scientifically reliable, it has to be repeatable. Repeatability implies that it be documented in some form, if only to allow people to make certain they didn't forget something. Of course documentation of the process may take the form of the report in which the examiner details what they did, and

344 See: http://www.ascld-lab.org/

345 National Association of Testing Authorities, Australia (NATA) "Forensic Science Accreditation Program – Electronic Evidence" - Technical Circular 9 – November 2008.

there is no mandate in the legal environment that says that an examiner cannot perform a task that has not been previously documented at some particular level of granularity. The legal system does demand that the process be reliable, but no specific reliability figures are provided by the legal system, and such a standard is unlikely to be produced by the courts.

Help files that describe the programs, process, or other elements of the effort are usually available and are very helpful in establishing the process and how it works. Manuals are also helpful, but the examiner and the courts typically understand that what a manual says does not necessarily track to what the actual system does. The same is true for checklists or almost any other form of documentation in common use. Documents can also be forged, altered, or the examiner can put in false or erroneous data. This is simply the nature of documentation.

Contemporaneous records of the process, and results consistent with the proper application of the process, help to establish that there is a process whose reliability can be tested and that the process was carried out as described. Notes taken at a later date or written from memory when faced with the requirement to document at a later time, tend to be less accurate. Some tools provide documentation in the form of audit trails or other similar records, and these can be very helpful in establishing, for legal purposes, what was done by the examiner, just as for the evidence.

In cases where specific types of measurements are done repeatedly, the process may be more precisely defined and additional details may be helpful, but at the end of the day, there is no defined level of required precision. The requirement of repeatability is part of science and is demanded by the courts in introducing scientific evidence, even if not for the precise sample in cases where the test destroys the sample. But in the DFE arena, the original traces are typically not destroyed in performing the tests. If examination results are challenged by opponents who have their examiner repeat the examination and whose examiner comes up with different results, better documentation and more precision is likely to be viewed with less skepticism, as is more expertise, experience, education, and training.

Tested components and tools

Components and tools used in examination should generally be tested prior to acceptance in their acquisition process. However, many of the tools that are widely used by examiners are commercial off-the-shelf technologies, like operating systems, personal computers, disk drives, and so forth; widely used utility programs, like the development tools that come with most Unix-like environments; open source free tools, like the Perl language; and hardware and support systems, like power supplies, keyboards, mice, and so forth.

As a result, there is rarely a formal acceptance process in the acquisition of the tools used for DFE examination, and the procurement process does not facilitate the sort of surety associated with commercial medical equipment, chemical laboratory equipment, or other similar equipment used in other sorts of testing and evidence processing. While evidence collection tools sometimes use this level of care, and NIST tests some such tools to verify that they operate as described, the same is not true of examination tools, at least not today.

The examiner therefore has to do their own testing of tools, and such testing is unlikely to be as thorough or complete as the sorts of testing done for many other sorts of tools. Test development is non-standard today, and complete tests for even simple functions of such tools is infeasible. However, some tools come with limited test suites as part of their implementation. Examples include the Perl computer language and the Clisp implementation of the lisp computer language. There are also many uses of common utilities in Unix and other operating environments on a daily basis, since these tools are used all the time all over the world. Again, these sorts of use-based reliability assertions and operational tests are not tests for specific forensic functions, but they are indications of reliability and suitability for normal business purposes. In this sense, they are as reliable as the business records relied upon by businesses for other purposes and allowed into legal cases as an exception to the hearsay rule.

There is extensive literature on testing of hardware and software, as described in Chapter 5. The examiner familiar with this literature may be at an advantage in performing such tests. To the extent that

these sorts of tests are used to validate a suite of tools and that the tests are retained for ongoing verification that the tools continue to operate as intended, this supports the fact that the tools are reliable for the purposes they are designed and tested for.

It is also fairly straight forward to produce tests based on a fault model and demonstrate that tools don't have any of the faults identified by the model at the level of coverage associated with the test regimen applied. For example, a test of the Unix "grep" command with the "-i" switch that ignores case differences can be built up in a simple way, such as by generating test cases for particular strings. Here is a simple test generation approach that tests for case sensitivity in grep:

```
for i in a b c d e f g h i j k l m n o p q r s t u v w x y z; do for j
in A B C D E F G H I J K L M N O P Q R S T U V W X Y Z;
do echo $i$j;echo $i$i; echo $j$j; echo $j$i;done; done | grep
[AbCdEfGhIjKlMnOp][QrStUvWxYz]

!!|wc
```

The first part of this should produce pairs of letters in sequence that include lower/upper-case, lower/lower-case, upper/upper-case, then upper/lower-case English letter pairs, where same case pairs are repetitions. This should come to 26*26*4, or 2704 "words" of 2-letters each, one "word" per line, ending with a newline character.

After this, it selects out only those pairs that start with A, b, ..., O, p and end with Q, r, ..., Y, z in a case-sensitive manner. This comes to 16 first characters for each of 10 different second characters, but excludes half of them because no same-case pair with different letters is produced by this generating algorithm. Thus we should see 80 2-character sequences. Output should be one 2-character "word" per line, with a newline separating lines, for 80 words, 80 lines, and 480 total characters.

The second command (!!|wc) counts words, lines, and characters from the result of the previous command, and the correct answer to this is "80 80 240" (lines, words, and characters), which is confirmed by running these commands.

```
for i in a b c d e f g h i j k l m n o p q r s t u v w x y z; do for j
in A B C D E F G H I J K L M N O P Q R S T U V W X Y Z;
```

```
do echo $i$j;echo $i$i; echo $j$j; echo $j$i;done; done | grep
-i [AbCdEfGhIjKlMnOp][QrStUvWxYz]

!!|wc
```

The first command here reproduces the first command above with the "-i" option to "grep", and the final command repeats that case-independent command counting the number of results from that command. Ignoring case, the answer should include each of the 16 first characters for each of the 10 second characters, in both upper and lower case, for a total of 160*2, or 320 total two-character sequences. This should produce 320 "words" on 320 lines, and at 3 characters per line, 960 total characters.

The second command (!!|wc) counts words, lines, and characters as above, only using the results of the second case-independent search. The correct answer is "320 320 960" (lines, words, and characters), which testing confirms.

The difference between the outputs from these two sets of runs comes from the difference between case dependent and case independent versions of the use of "grep". As tests go, this test only models a very limited sort of fault set, but it gives some level of confidence about this function of this program.

To the extent that simple tests like this one can be rapidly generated to verify some properties of tools relative to the case at hand, they may be worth doing as a validation of operation. For example, in a case where specific strings are being sought within a particular trace, those strings could be used for the test and a test case where those strings are inserted into a file without those strings previously encountered would be reasonably convincing. It might be all the more convincing if the file was of the same type as the one under examination, and so forth.

A more comprehensive test set is clearly called for in the testing of such programs, but without a fault model that suits the need, the development of such a test set will only be of limited value. This is why the alternative strategy of using redundant tools provides a means to verify results and assert reliability of results even if tools themselves are imperfect, and for cases where we cannot calculate

the "right" answer in advance. For example this result can be verified with the following test:

```
for i in a b c d e f g h i j k l m n o p q r s t u v w x y z; do for j in A B C D E F G H I J K L M N O P Q R S T U V W X Y Z; do echo $i$j;echo $i$i; echo $j$j; echo $j$i;done; done | awk '/[AbCdEfGhIjKlMnOp][QrStUvWxYz]/ {print $1}'

!!|wc

for i in a b c d e f g h i j k l m n o p q r s t u v w x y z; do for j in A B C D E F G H I J K L M N O P Q R S T U V W X Y Z; do echo $i$j;echo $i$i; echo $j$j; echo $j$i;done; done | awk '/[aABbcCDdeEFfgGHhiIJjkKLlmMNnoOPp]
[qQRrsSTtuUVvwWXxyYZz]/ {print $1}'

!!|wc
```

This program uses the "awk" program to perform the same functions done in the previous example by the "grep" program, so it should produce the same results, and it does. Of course the "wc" program could be making the same error in each case, or the "awk" and "grep" programs might use the same underlying libraries that are in error, and so forth. Other more separate and different test cases can be used to provide higher assurance, and more test cases may be generated to match different circumstances.

Size matters in DFE examination. Part of testing regimens should account for large traces, large numbers of traces, overflows of counters, the time and space complexity of the technique being applied, and other sorts of issues typically associated with such tools. For example, using a similar sort of testing approach:

```
for i in `count 1 1000000`; do echo "AA"; done | grep "AA" | wc
```

In this case, the "count" command is simply a program that counts integers, in this case, from 1 to 1,000,000. It should generate 1 million words and lines, and 3 million characters. The output confirms this, and the test, on a system used for this test, took just under 20 seconds to complete. To repeat this test for a billion samples will take 1,000 times as long, or about 5 hours, and for a million times as long, it will take 208 days. This is, of course, a very

simple test, and to do such a test for all patterns of symbols of a given length would take far longer. In practice then, the examiner must determine what tests will be relevant to confirming the operation of the tools in context. At some point this tool, like any tool, will fail. The issue for the examiner is, at what point it fails, and how does that impact the case at hand.

The test above failed when the count was set to 100,000,000, but this does not mean that this is a failure in "grep" or "wc". Tools execute in the context of their operating environments, and it is incumbent on the examiner to test their tools in the environment they operate within in order to have meaningful results. This particular failure produced error messages indicating that the failures were related to memory exhaustion. That likely means that in an operating environment with more memory, the test would have gone further, or that a file-based process would have been able to go further. As one limitation is overcome, another limitation will become the first to be exposed. Ultimately, these limits will have effects on results of examination, unless the examiner is aware of and compensates for the limitations, wherever they may lie.

Keeping independent things separated

Laboratory methods in common use in science apply different approaches to assure the purity of sources, results, and processing mechanisms. This includes a wide range of activities, depending on the different sources of contamination. For DFE, separation can be kept far more easily than for chemical effluents, blood splatter, or other similar things. But separation and proper cleaning are still required in order to get accurate results from many tools.

Contamination of content in digital systems comes from a variety of mechanisms. Examples include, without limit:

- **Failure to erase previous results before reusing the same file or directory for new results.** In many cases, results are appended to previous results, the same file names are used, or the mechanisms that the tools use don't remove all residuals from previous executions. This can produce a form of cross-contamination where results from different runs are mixed, duplicated, or overwrite each other.

When then used by subsequent tools, the results will appear to be normal but will not be right.

- **Failure to compensate for identified failures or errors in the use of a tool.** In many cases tools will produce error indicators that are not seen because they scroll past the screen before the examiner can see them or they are routed to an error log file that is not examined. In these cases, the errors may or may not produce wrong results, or place the results in a wrong location that is used for other purposes. The examiner should capture and examine such error messages in order to properly understand the results produced.
- **Failure to validate results of one step before going on to the next step.** In many cases, an execution of a program will produce wrong results because the examiner makes a syntax error or some other similar mistake in the use of the tool. Subsequent tools that are executed later in the sequence may depend on these results, and invalid results from a prior step may result in wrong execution of subsequent tools which may produce results in the wrong place or overwrite or artificially augment other correct results, and so forth.
- **Cleanup processes may remove results that should not be removed.** Sometimes filenames that are not expected are produced from an analysis, and the incorrect removal of those files by a tool may produce wrong or missing results.
- **Accidental saving over content that should be read-only.** It is common for examiners to examine a result, intermediate file, or even source information, using an editor that is capable of modifying files, relying on their skill to not save any changes. It is obviously far better to write lock such files prior to use, but this is often missed in the process, and may result in saving a modification that should not be saved.
- **Files left as the result of failed processes.** Many tools create and clean up temporary files as part of their operation. In some cases, a tool failure or interruption results in temporary files not getting deleted. These temporary files

may end up being reused by another process or producing multiple or erroneous counts or other similar contaminations.

The list of possibilities may be endless, but the lesson is simple. The best way to prevent such contamination is through separation. The problem is that separation gets more expensive as it gets more certain, and at some point, it is more of an impediment than a help.

Systematic approaches to separation typically consist of the creation of barriers between things, ranging from separation of the forensic examination network from other networks, to separation of cases from each other in different systems, disks, partitions, or directories, to separation of inputs, processing, and outputs from each other and from other processes. To the extent that there is a method used for separation, it helps if it can be explained or is documented, and it helps if it is followed and/or enforced by some well controlled mechanisms. This is similar to access controls, but at a more diverse set of levels and at finer granularity. And this is where the problems begin. Because systems for examination are not built for the purpose, automatic controls that might be effective for this purpose do not generally exist, or are hard enough to manage that they aren't very useful in practice and interfere with meeting the requirements of schedule.

Most DFE examiners today rely on technique and their own skills, training, knowledge, expertise, and experience to prevent or mitigate such separation errors. They may leverage tools such as write protection settings, use the directory structure to keep things in groups, such as having files or directories separated by date of reception or processing, or they may simply be careful about what they do. There are also sometimes checks performed after steps to identify process failures and repair and redo the failed examination process or step. These steps sometimes detect contamination issues as well.

Assuring the purity of original and duplicated evidence

As a rule of thumb, the DFE examiner should, at most, only come into contact with original writing at its initial duplication. After that point, it should normally be physically secured somewhere else, and never be brought into contact with any examination system.

DFE examination normally takes place starting with traces that are exact copies of original traces (O-traces). If these traces are not accurately reflective of the O-traces, the results of examination are likely to be wrong, and perhaps more importantly, the reliability of the processes and methods are questionable. Because exact copies of DFE can be easily made, there is little excuse for a process that fails to retain original evidence. There are; however, good reasons to use derived traces (D-traces) for analytical processes, because the analytical processes may be more efficient or may allow the use of reliable existing tools for many purposes by using D-traces in place of O-traces.

It is probably wise to keep backups of O-traces, to store the ones that don't take excessive space on write-once media, such as a CD-ROM, and to use cryptographic checksums or a similar mechanism to allow the O-traces to be checked for alterations at a later date. Write lock mechanisms within operating systems are also helpful in reducing accidental alterations, and working from copies while keeping originals in a separate location helps from a separation standpoint.

Validating purity of duplicated and derived evidence

In making copies of traces, it is a reasonable and appropriate, but not a necessary step, to confirm that the duplication was properly done, by performing cryptographic checksums, byte-for-byte comparisons, or verifying characteristics such as length and other similar data to the originals.

When generating D-traces, which is often done in cases where a trace includes subsequences analyzable as separate sequences within the overall context of a larger trace, it is also helpful to verify that the subsequences are properly extracted from the O-trace, so that use of those traces does not introduce errors into the process that would not be present in the O-trace.

As an example, when taking a message collection and extracting the entries indicated by the syntax of the collection mechanisms, checks that can be easily performed include verification of the count of messages extracted, examination of the first and last against the O-trace to make certain that the first and last message were included in the resulting D-traces, counting the number of

headers, separators, and bodies in the original versus the number resulting from the derivation, and so forth.

This part of the process helps the examiner to validate their results, and errors in these processes that are not caught, may result in faulty examination results that destroy the credibility of the examiner and their results. On the other hand, there are limits to the ability to do this. For example, if the same process is used for every step at every level of granularity, the examination process would likely collapse under the weight of the verification process. At some point, a balance must be struck between schedule and certainty.

Known test samples with known results

Using the same tests or sorts of tests used to verify properties of the tools, tools can be verified just prior to use or at other times by the use of verification suites. The creation of verification suites provides for ongoing confirmation that tools work as they should. Such tests are commonly undertaken when a tool is part of an upgrade or an update to a system, when there is a patch installed, or in other similar cases where there is the potential for a tool change that might affect examination results.

In these cases, a common approach is to create samples with known properties, verify those properties across multiple systems, and use the known samples and known results to verify that tools work as golden units work. The testing consists of running known samples with the tools under test and confirming that results match results from golden units. To the extent that all tools and environments have breaking points, the examiner compensates in evaluating the results. When tests fail, the examiner who wishes to use these tools has a responsibility to understand the limitations identified and not mistakenly use a result that is not valid.

Contemporaneous notes

While processing DFE, the examiner should normally take notes of actions that are relevant to the matter, and in particular, of any examinations that are going to be used in the case. Unfortunately, this is not always feasible. For example, in criminal cases, defense counsel will often tell the examiner not to take any notes because such notes may provide evidence that can be used against the

defendant, or that can be used to question the examiner about their processes.

Such notes, and all working papers of examiners on all sides that are going to appear in court as expert witnesses, are discoverable under most current legal regimens. If such notes include opinions, expressions of frustrations, jokes, or simply a list of things tried that were not revealing, they may be used by the other side to make claims about the examiner's competence, technique, and so forth.

Legal counsel may allow the examiner to use whatever process they normally use for taking notes or writing reports, and laboratories commonly have standard approaches for taking notes on processes undertaken in examinations.

One approach identified[346] is to always work in the context of writing a report, and take contemporaneous notes within the body of that report, along with recording relevant results of examinations. Where results are voluminous, they may be recorded in files and identified within the draft report. As the case proceeds, the report is updated. Old copies are overwritten with the new version to eliminate errors associated with accidentally looking at an older version of the report, which has been updated to correct for previous errors. By the time the report is ready to become finalized, it includes contemporaneous notes on activities performed at various dates and times, along with relevant results. But any speculations that didn't pan out, or other similar sorts of items, are removed in forming the final report, without special effort and without a destruction process.

In any case, technical aspects of DFE examination cannot realistically be remembered without notes at a level of detail required in order to accurately report them out. As a result, some level of note taking is almost always necessary and appropriate, and such notes should be taken as the activities are performed. In cases where the activities are done separately, it is prudent to redo the activities while taking notes to assure that no errors appear in reports and that the process was properly followed and can be repeated. As an example, in this book, all of the examples that

346 F. Cohen, "Challenges to Digital Forensic Evidence", ASP Press, 2008 ISBN#1-878109-41-3.

include the execution of commands on a computer were done at the time the writing was done, and the exact commands and results from those times are included.

Calibration with known samples prior to use

Calibration is normally undertaken with known samples as described earlier. In chemical and other similar processing, a standard practice is to:

- Do a calibration with known levels
- Do a cleaning process to remove residuals
- Do a test to show background levels
- Perform the test on the sample under analysis
- Perform a cleaning process to remove residuals
- Perform a background level to confirm background levels
- Do a calibration with known levels

This process allows verification after testing to confirm that the testing mechanism is still calibrated and that background levels are consistent before and after the tests.

In the DFE examination arena, this is typically not necessary or appropriate. For one thing, in DFE examination original evidence is not normally being used. Rather, forensically sound copies of the O-traces are used. For another thing, bits are either 1 or 0, and thus much of the background testing and calibrations testing is irrelevant to the two-state world of bits that applies to DFE. Nevertheless, there is little cost or harm in doing simple calibration tests prior to use, such as through the use of the sorts of test identified above.

In performing such tests, it is particularly important to make certain that any results and residual data produced during the tests are cleared prior to use on the O-traces. Otherwise, the testing process itself may leave residuals that cause the examination results to be corrupted.

Again, this is an area where the DFE examiner's knowledge of systems and mechanisms used in performing the examination is key to doing a proper job.

Use of tools consistent with procedures

The use of tools should generally be consistent with procedures to the extent that procedures exist for the use of those tools. However, most examination other than some of the more rudimentary elements of analysis with off-the-shelf tools, requires that the examiner create specifications, connect together tools within the operating environment, or program tools for the specific issues at hand. As a result, the only real specification for tool usage tends to be the help entries and manuals that describe those tools and how they work.

In most non-trivial examinations today, tools are put together for the purpose using overall techniques and methodologies that are well understood, but still in a manner that is customized to some extent to the particular situation at hand. That means that the use of these tools has an internal generate, test, apply, verify, and correct loop.

As an example, suppose the case demands that the examiner confirm or refute the consistency of traces in the form of audit records with events that assert that the user identity "Joe13" always logged in within 25 minutes of the execution of one of three programs, each of which subsequently sent files to one of five different outside computers via the Internet. In such a case, there is no tool that is designed, configured, and operated so that the examiner enters that sentence and the results pop up, and there is not likely to be such a tool in the foreseeable future. Rather, the examiner must interpret this by creating a set of hypotheses that are independently tested against available traces, and this may be done by doing a series of analysis processes using tools. For each of these steps, the examiner should normally go through the loop.

Here is an example of a possible first step in examination for this case:

- **Generate:** The examiner generates a small program using existing and tested tools to extract the records of user logins with the user identity Joe13.

- **Test:** The examiner runs the code on a sample of the traces to see if it is generating the right answers in the right format. Until this is correct, the examiner will loop back to generate and test. Note that this may clear up many errors, but may leave many as well.
- **Apply:** The examiner applies the, now tested, code to the traces as a whole to generate the output, and likely stores the output in a file.
- **Verify:** The examiner reviews the results to verify that no unexpected cases came up, and may selectively look for cases in the O-trace to verify that the analysis worked as intended. The examiner may include a range of tests including redundant processes to confirm that these results are correct or refute them.
- **Correct:** If the results don't confirm properly the examiner makes corrections to generate a new version of the program and continues from there.

This is the normal process to be expected in examination. When a similar examination is done at a later time, many of the previous little programs may be reapplied with minor changes, but the same steps will likely be applied if a sound result is to be found. In the process of performing this effort, the examiner will typically record the final tool use and results in contemporaneous notes, but it would be a waste of time, effort, and space to include every step along the way. For example, a search that fails because of an error in the specification of a search term might result in gigabytes of useless results. To keep them would be problematic at best.

Checking results with redundant process after use

As the example above shows, redundancy is commonly used in the process of examination to verify that results of examination steps are not obviously incorrect.

Another very common step is to check results after intermediate and final results are produced, and do a verification of things like counts produced against expected counts, do sanity checks on results, examine specific instances to confirm that what was found was expected, and so forth. To the extent that these things vary

from expectations, the examiner will then adapt the process and retry those processes to get them to work properly with the traces under examination. This often identifies tool limitations that are then worked around by the examiner.

This step is part and parcel of the internal loop above, but even after a successful examination is completed, such redundancy over the entire process if desirable, particularly if the result is critical to the case or will be reported out or presented in court, rather than just used to help guide further examination.

Presentation tools and visualization

Examination uses presentation and visualization to see the results of other tool uses, and to the extent that the examiner depends on these presentations, they are critical to examination outcomes.

Because DFE is latent in nature, visualization is essentially always required in order to deal with examination. For example, getting the results of the counts of characters, words, and lines from the "wc" program requires that the examiner read the output on the display. The bits that are represented in traces are not directly visible, nor is there a single common representation used to examine even bits in files of defined types. There are multiple methods used to view results depending on what the examiner wants to see about them.

The visualization tools used by the examiner determine what the examiner sees, and as a result, these tools must be treated in the same manner as any other tools used in the process in terms of assuring their proper function in context.

Things tools don't show well or at all

Even the depiction of sequences of bits and bytes is problematic to the examiner because different tools show different depictions of otherwise identical sequences. For example, and without limit:[347]

347 F. Cohen, "Fonts For Forensics", IEEE SADFE (in conjunction with the IEEE Oakland Conference), 2010-05-19, Oakland, CA.

- In viewing a file that contains characters from the ASCII character set, a file containing characters represented in EBCDIC will not be properly visible.[348,349]
- In viewing files in ASCII from an editor designed to edit ASCII, some characters, such as non-printing characters, may not be displayed at all.
- In viewing unicode files there may be different character sets that are displayed differently in different viewers and indicate different results or meanings.
- In viewing characters with different bit representations, the displays may be so similar that they cannot be told apart by typical inspection.

Different tools indicate different results, or may have special options for viewing things in different ways. It is surprisingly easy to miss parts of sequences of bits that may be important to the matter at hand.

Examiners may get results from one tool that appear to disagree with results from a different tool because of such visualization results. For example, in using the "diff" program to detect and display differences between two files, differences may be indicated where none are seen when viewed. This may seem inconsistent until the viewing is adjusted to see normally unprintable characters, like "^G" (ASCII 7), which normally displays as the ringing of a bell, when the system includes audio output, or may be displayed as a flash of the screen if visual bell is enabled in some terminal output modes. By viewing the line through the "less" program or using the "hexdump" program, different results will become visible.

Fortunately, DFE has finite granularity, and as such, all traces can be examined at the level of the bit. This is what the "hexdump" program does. At the same time, this is of little help in understanding what a graphical image will look like, because hexdump does only very non-graphical presentation of the content. Different tools present the same bit sequences in different formats,

348 The American Standard Code for Information Interchange (ASCII), standard X3.4-1963, American Standards Association, June 17, 1963.

349 Extended Binary Coded Decimal Interchange Code (EBCDIC), for details see: http://www-01.ibm.com/software/globalization/cdra/appendix_a.jsp

and the examiner must select the visualization tool depending on their interpretation of the type and the utility of the visualization for the legal matter.

Validating visualization tools, even for text display, is problematic in that any testing will depend on human cognition, and it is very hard to get precise answers or do high volume tests in this situation. Rather, the examiner must have knowledge of the tools and their limitations and use redundant methods to try to validate their own results. Different examiners using different tools may come up with different answers from such visualization differences, and there will be little that can be done to mitigate this problem other than careful practice by the examiners.

Forensic Fonts

As a general rule, it is highly desirable that displayed symbols from a defined symbol set used for legal purposes be precise, accurate, and unique. Precision and accuracy of representation are well understood in the legal community and, for the presentation of scientific and technical evidence, have been highly supported by legal rulings. The uniqueness property is highly desirable to avoid confusion and allow definitive answers to be given to specific questions that may arise. As a first attempt to characterize a set of rules and basis for those rules when devising fonts for use in forensic examination and presentation, the following criteria, as identified earlier, were asserted:[350]

- Each symbol should be clearly different from other symbols.
- Each symbol should be familiar, with minimal interpretation, so that it looks similar to what normally might appear.
- Each symbol must be printable so that a <space>, <tab>, <carriage-return>, <backspace>, <escape>, and other "non-printable" characters can be clearly seen on printed pages.
- Each symbol should self-indicate the bit pattern that produced it so that it can be traced back to its original value.

350 F. Cohen, "Fonts For Forensics", IEEE SADFE (in conjunction with the IEEE Oakland Conference), 2010-05-19, Oakland, CA.

A side effect of these criteria is that the font will take up more space on a page than the normal font would take up for the same level of readability, and it will have some differences from the fonts commonly used for other purposes, such as a more distinct difference between 0 and O, 1 and l, l and |, and so forth.

The example from Chapter 6 uses the forensic font™ tool to show output that is otherwise indiscernible that becomes clearly understandable with such a font. Figure 9.1 shows a simple shell script that allows a forensic font™, represented by a series of graphical image files, one for each symbol (byte), to be produced for viewing with a Web browser.

```
echo "fixer [FS] S0A.jpg test 0A 12 32"
echo "   Full or Small - input file - output file (.html) break [width [height]]"
Font=$1;shift;InFile=$1;shift;OutFile=$1;shift;Break=$1;shift
if test "X$1" == "X"; then WL="";
  else WL=" width=\"$1\"";shift
  if test "X$1" == "X"; then WL="$WL"; else WL="$WL height=\"$1\""; shift; fi;fi
for i in `hexdump -v $InFile | toupper |
          while read a b; do echo $b;done`; do
               echo -n "<img src=$Font$i.jpg hspace=\"0\" vspace=\"0\"$WL>";
          if test "$i" == "$Break"; then echo "<br>"; fi; done > $OutFile.html
```

Figure 9.1 - A simple forensic font™ tool for use with a Web browser

The result is an HTML file that looks something like this:

```
<img src=ASCII/F31.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F2C.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F34.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F63.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F31.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F2C.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F34.jpg hspace="0" vspace="0" width="16"
height="36"><img src=ASCII/F0A.jpg hspace="0" vspace="0" width="16"
height="36"><br>...
```

This then displays through a Web browser. Similar output using a more advanced tool with essentially the same functions is shown in Chapter 6 under “What you see is not what is there”. Regardless of the mechanism used to print such output, as long as all of the output is shown, a DFE examiner who knows what a forensic font™ is, should have no difficulty in identifying what is present at the bit level in the item being displayed in this fashion.

Going faster using the visual cortex

In processing DFE, examiners often use visualization to allow them to make decisions more quickly. This is because the visual cortex does automatic processing of visual images very quickly. The sorts of things that it does, include, without limit, detect lines, interpret meetings of lines, detect surfaces, assemble parts into larger structures, recognize faces, and differencing operations between images.

While there are increasingly programs designed to look at images and describe them in words, these sorts of analysis are not very good for searching images for depictions of individuals doing different things, differentiating a child from an adult, automating analysis of shadows, or any of the many other things that DFE examiners might do in a legal case. They are also not very good at doing precise analysis.

This means that the DFE examiner can be very efficient at seeing similarities between things visually that might otherwise require writing a substantial program to do. For example, in comparing sets of headers of packets to one another to see patterns, automation is good at things it is programmed to do, but in looking at a time sequence associated with a particular destination IP address, the rate of traffic flow of different types is readily understood when depicted as graphics, and a human examiner can see correlations between different sorts of packets visually with substantially less effort than it takes to write a correlation analysis program that looks for the same thing.

After the examiner identifies this pattern, they can extract the relevant parts of the traces using different tools and put them together in any number of different formats for analysis and reconciliation with other activities, such as log files. There are different tools for this, but without the visualization by the examiner, picking out the pattern of interest requires tools that don't exist making judgments that tools do not yet have the capacity to make.

Cognitive errors and visualization

A substantial danger exists that the cognitive systems of the human interacting with the visualization of the computer may produce joint errors that are far worse than either would produce on their own.

Current computer programs are not generally designed with understanding of human cognitive limits, and as such, they produce a wide variety of outputs that humans may be unable to take advantage of or may misinterpret.

An example of a serious potential gap is that human eyes respond to different colors differently, but displays produce identical outputs regardless of who is looking at them. A person who has some level of color blindness, for example, might completely miss patterns that involve certain colors. Similarly, small variations in colors are very hard for people to distinguish at the level that modern video displays distinguish them.

A very different example comes in the form of the high degree of display resolution and color depth of modern displays preventing people from visualizing things that might be better seen at lower color depth or resolution. For example, at high resolution, edge lines appear to be slowly changing gradients, while at lower resolution they are more distinct. A person might miss such edge lines at the higher resolution. A side effect of the exaggeration of differences near a discontinuity and suppression away from the discontinuity might be that a human viewer at a lower color depth would see content within images that would be missed by the same person at a higher color depth. Some things actually become more readily visible with less color depth.

There are any number of other sorts of cognitive errors associated with visualization, and the student of these issues might find a great deal of information elsewhere.[351,352]

The need to understand the tools and processes

If one thing is clear from the discussion of tools and processes, it should be that the DFE examiner needs to have a sound understanding of the tools, the traces, the operating environment, and how they interact, in order to produce sound results.

This level of understanding is very difficult to attain for many of the commercial products on the market that operate only through

351 Donald D. Hoffman, "Visual Intelligence: How We Create What We See", Norton, 1998, NY.

352 Al Seckel, "The Art of Optical Illusions", Carlton Books, 2000.

graphical interfaces and provide little in the way of specific details. These tools are almost impossible for the examiner to test in the same way as tests were demonstrated herein, and as a result, the examiner is left largely trusting whatever the tool produces. Redundant methods are rarely available with these tools, displays are controlled by the tools, and this limits the ability to use different visualization methods.

Reconstruction uses commercial or open source tools, as do most other examination processes, but these are typically constructed by the examiner by combining different parts together that each have other uses than the forensic examination process. As the complexity and sophistication of those component tools goes up, so does the requirement for the examiner to know more about them and take more care in their use.

Many of the most experienced examiners that do complex cases build their own tool collections and do their own testing, so when they need to change a tool or validate a result, they have the chance to do so independently of any particular vendor. Others may use tools from different vendors and do redundant work using the different tools to achieve some level of certainty above what a single tool can give them.

Most people who do examination today are only taking part in a limited portion of the analysis function and use only a few tools that they don't fully understand. They tend to work on standard cases, such as searches for images of one sort or another, identification of obvious documents and printout of their contents, or examination of log files for specific items indicative of only a limited set of issues.

As the examination process becomes more complex and requires work that is beyond what current off-the-shelf tools provide, the expertise level of the examiner must go up.

Creating and using a "golden unit" environment

Very few individuals in DFE examination end up going to the extreme of building or assembling their own DFE examination tools or tool collections into a cohesive set of mechanisms held together by some unifying framework. When they do so, they tend to create some form of a "golden unit", which can be reproduced and reused

from case to case. It may be improved with time, but if packaged as a tool or set of tools, the collection becomes a unit in and of itself. In some extreme cases, these individuals have built their own bootable operating environments in which they have characterized their tools and configured the environment for forensic examination.

One such example is the "White Glove" bootable Linux CD[353] that the author of this book created some years ago. This includes the "ForensiX" forensic examination software,[354] and is now available for free in source form from over the Internet. Today, there are many similar tools, each asserting and implementing various properties, some of which may be desirable for doing DFE examination.

Creating the operating environment

The environment in which tools operate is key to reducing the time, effort, and cost associated with validation and verification. This "golden unit" software only runs in a relatively controlled environment, testing can be limited to that environment, and configurations of the environment can be standardized to support the DFE examination function rather than be designed to operate in other environments.

The White Glove version of Linux (WG) comes on a bootable CD-ROM. The CD-ROM can be duplicated and a copy kept with each collection of traces and results for each case, so that the precise software and operating environment used can be reproduced for validation and independent repetition of all activities. This means that for each case, at the cost of about $1, a copy of all the tools actually used can be kept, and this makes subsequent challenges far easier to manage and results easier to reproduce.

When WG starts up, it looks for and finds hardware, but it mounts all of the disk drives read-only, does not assign an IP address to network interfaces, but allows them to operate as passive sniffing devices, and generally works to preserve the integrity of all content on the system. This provides some assurance that only the actions taken by the examiner are done by the operating environment.

353 See http://all.net/WG/index.html
354 See http://all.net/ForensiX/index.html

WG does not start with many background processes running. In fact, it only supports user login on the console(s) and basic background functions required for the operating system to function. Every other function that gets enabled gets enabled by the examiner so as to prevent accidental alteration or activities that might be out of the control of the examiner.

Within WG, there is a read-only Web server that can be started up and which contains the manuals for all software contained on the CD, so that it is self-documenting and the version of the manuals for the version of the software contained are all on the CD-ROM. It also has a simple graphical interface that the user can enable with a single command, and other similar sorts of functions that operate in a default mode upon a single user command. For example, the examiner can write enable a disk with the "we" command and write lock the disk with the "wl" command. All of this is designed to provide convenient functions that support the examiner and provide protection against common errors by default. WG is also configured to handle disks of 4 Tbytes in size and files of the same size, so that if used properly, space limits are not usually problematic.

Tools within WG

Within the WG environment, and in most Unix-based systems today, there are many tools that are configured to work within the system. In the case of WG, this includes a wide range of commonly available utilities, tool kits, programming languages, editors, format conversion tools, and both widely used and WG-specific forensic and investigation-focussed software, all configured to operate within the environment and tested to some extent within that environment. There are also user interfaces for many of these tools that are designed to make them easy to use for the most common forensic uses, which tend to be somewhat different from the uses for non-forensic purposes.

Some of the tools are integrations of perl scripts, lisp programs, and utilities, or other such combinations that perform functions that would otherwise have to be programmed by the examiner. Over time, while WG was being maintained on an ongoing basis, tools were added as they were developed. Since the WG bootstrapped from a CD-ROM, the tools used for any particular case could be

kept in a forensically sound manner for and with that case, while updated versions could be used on the next case, or if multiple versions were used on a single case, multiple CD-ROMs could be included in the case file.

The next generation golden unit

Today, there are many surviving bootable Linux CD-ROM operating environments, and any of them can be leveraged for a similar purpose. There are commercial bootable CDs that have forensic functions, and there are many bootable versions of Linux that can be converted or customized for the purpose. While the operating environments will likely be somewhat less stable than one designed specifically with forensics in mind, they still offer many of the advantages of bootable media in terms of reproducibility of tools and results and inexpensive retention of specific tools used with the case file. As better testing becomes more common, such tools will likely become even more reliable and have better known limitations that are documented with the tools so the examiner will better be able to understand their limits and applicability.

Virtualization is the next logical step in the use of "golden units". Instead of using a physical CD to boot up the system used to perform forensics and dedicating hardware to each forensic task, virtual computing environments (or virtual machines - VMs) with different configurations, tool sets, and capabilities are already starting to appear. This is similar to the concept emerging for reconstruction as described earlier, except that each VM acts as a clean room of sorts. Separation of cases and portions of cases is between VMs rather than between files, and this allows a shared facility to be used for many simultaneous forensic examinations with greatly reduced risks of cross-contamination. It also provides a common set of testing environments for tools, which can be shared across many processes without the cost or complexity of managing each environment separately.

As shared VM environments begin to grow and standardized tool sets with well-tested capabilities start to arise, input from one VM will ultimately be fed to another VM for subsequent processing, allowing combinations of tools that operate in different operating environments to be used for complex forensic examination with less

overhead and better ease of use. Specialization may also allow more people to work collaboratively in such environments.

Toward automated analysis and processing

As a central theme in modern forensics research and development, new methods and tools to implement them are first developed by researchers, tested in a laboratory setting, and implemented for free or licensed distribution by the researcher.

Over time, these tools either fall into disuse, are used by the researcher and a small number of other high-end forensic examiners for special purpose tasks, or become part of the overall corpus of methods and tools implemented in larger scale commercial offerings.

Examples include the tools identified above, and a wide array of other similar tools, including, without limit:

> FACE "a tool for automated evidence discovery and correlation"[355]
>
> PyFlag "advanced network forensics framework"[356]
>
> An email authorship attribution tool[357]
>
> Email Tools "A toolset for facilitating analysis"[358]

These tools undergo various levels of peer review, scrutiny, testing, validation, and application in legal matters, and to some extent, they represent the state of the art in digital forensics tools. And yet none of them can reasonably be trusted without independent validation of results by the DFE examiner using them.

They key to understanding what these tools have in common, is that they are "moving up" the analytical chain, from simple tools that do search for standardized patterns, to more complex

355 A. Case, A. Cristina, L. Marziale, G. Richard, and V. Roussev "FACE: Automated digital evidence discovery and correlation", Digital Investigation 5(2008) S65–S75.

356 M. Cohen, "PyFlag – An advanced network forensic framework", Digital Investigation 5(2008) S112–S120.

357 M. Corney, "Analysing E-mail Text Authorship for Forensic Purposes", Masters Thesis, Queensland University of Technology, March, 2003.

358 F. Cohen, "Attribution of messages to sources in digital forensics cases", HICSS 2010.

analytical tools that perform more advanced sorts of interpretation, attribution, and reconstruction.

Over time, it is to be reasonably expected that these sorts of tools will continue to be developed, and some will remain niche tools for experts, while others will be packaged into high volume forensics software applications and combined with other tools to form more cohesive packages.

Individually, they provide limited value, but in concert, they create a corpus of methods with reliability that can be reasonably explained, and as more and more of them are used in any given case, their effect become cumulative. While a single refutation may win or lose any particular case, the methodologies being developed appear to be sustainable, if properly applied, across many cases.

Questions

1. The National Research Council report appears to indicate that the DFE examiner should do something that we don't really know how to do well at this point in time. What is this thing, and how do we accomplish the objective?
2. Given the problems with peer review and its limitations on determining the validity of scientific methods, how much of a confirmation is a peer reviewed article?
3. Is widespread use of a software mechanism sufficient to make its use reliable for forensic purposes?
4. Suppose a whole list of different approaches have been used to count the number of instances of some string within a larger string, and all agree that there are 45,872 instances. Is there any way to be truly definitive about this count? If so, how? What if one of these methods indicates 45,871 but matched all of the other methods in its results to all test cases? How reliable would either of the counts be and how would we reconcile them? Suppose we knew of 50 different ways to make the count but only tried 2 of them. How would we know that the next 2 wouldn't produce a different count?
5. Given the answers from the previous question, how should the limitations of things as simple as counting or searching be identified to the court in reports and on the witness stand? What can the examiner really say about the output from tools where the results cannot be manually verified?
6. What is the difference between validation and verification? When should which be used?
7. How does the DMCA limit the ability to attain precision in reconstruction? What can the DFE examiner do to mitigate these limitations?
8. When using commercial closed-source tools, how can the DFE examiner validate them or understand how to test them given the DMCA and the lack of access to source code?
9. Process controls can become a nightmare of paperwork, especially when the DFE examiner has to define each

process for each activity they perform. Is this likely to be a temporary condition until the full range of examination processes have been thoroughly covered, or will this remain an issue indefinitely because each examination will continue to be unique based on the specifics of each case?

10. Given that the current accreditations don't cover most of the aspects of examination as defined in this book, what good is such an accreditation in terms of doing such examinations?

11. What sort of commonalities will defined processes have? How hard is it to create a standard approach to defining processes for documentation purposes?

12. Given that examination uses common tools and combines them together in many cases to do tests, is it adequate to test the tools? If not, what are the tests for combinations of tools and how are such tests defined? How can all interactions between tools as composed be defined, understood, modeled, and tested? Or is there another approach that will work better for combinations of tools?

13. Create a test to determine whether "grep" was the cause of the failure in the size test in this chapter, and run that test to determine the real limits of "grep" and "wc". What are the limits?

14. Identify a procedure for separating things in the computing environment you use for examination. What are the potential failure modes of your separation mechanism, and how does its use impact efficiency of the examination process?

15. Given that derived traces are not identical to the O-traces they come from, how can you validate the translation and verify that it is properly done for each case in which you use it? If the derived traces are not properly reflective of the O-traces, how can the results be right? If they cannot be validated against each other, how can you be sure the results are right?

16. Create a mechanism to generate test cases for determining the maximum integer values that can be attained by "wc" or a similar program you use in counting things within traces.

What is the maximum value? How could this create problems in examination results for large data sets? How will you compensate for these errors in your examinations?

17. Given the benefits and problems with contemporaneous notes, how will you take them? Document this as part of the process document for your examination processes. How will this documentation help or hurt with legal challenges?
18. Explain why DFE examination should not follow the same calibration process as chemical or other similar scientific measurement processes.
19. Given that examiners make mistakes and try to correct them in the normal process of examination, how can we tell that the things they finally decided were not mistakes weren't in fact mistakes as well? Does redundant process really solve this problem? Does it make it less likely? What about common mode failure mechanisms? How can this be put into a numerical value and presented to the courts? What is the validity of such a value?
20. Given the limits on presentation and human interpretation of differences in presentation, how can the examiner ever be certain of the results they see from their examination?
21. How can the examiner be sure that they are looking at the correct representation of traces? How can the examiner be certain that they see all of what is actually presented?
22. What cognitive errors impact visualization and how can they be avoided or compensated for by the examiner?
23. What sorts of problems does the forensic font™ approach solve, and what sorts of problems does it create. Can similar approaches be used for less obvious cases, such as non-text files? How can this be done?
24. Other than examiners that create their own forensic examination environments and program and validate them on their own, what is the best we can really expect from examiners in terms of their understanding and use of tools and processes?

10 Today and tomorrow

If the field of digital forensics is to be a scientific field, then scientific methods and systematic use of language are appropriate. DFE examination today is still in its infancy, and as a scientific endeavor, it is often lacking in methodology, rigor, and even in the way it is presented.[359.360]But this will change with time.

Today

The current state of consensus in the DFE examination community appears to be limited. In 2010, two approaches to identifying such consensus were undertaken,[361] and a follow-on study in 2011 resolved some of the limitations identified with the 2010 study.[362]

One 2010 approach was a survey in which a series of statements were posed to members of 4 different groups within the DFE examination community. They were asked to indicate “I Disagree”, “I don't know”, or “I Agree” to each of list of 14 simple statements, 3 of which were control statements from normal physics (two widely accepted physics truths, one a made up false statement). Of the statements relating to DFE examination, consensus above the margin of error for random guessing was present only for 2 candidates; (1) “It is possible to duplicate digital information without removing it.” (75%) and (2) “Computational complexity limits digital forensic analysis.” (64%). Among those rejected from consensus were several statements from the physics of digital information, including “It is possible to observe digital information without altering it.”, “Digital evidence is trace evidence.”, and “Digital evidence is finite in granularity in both space and time.”. All of these are truly fundamental to the introduction of digital forensic evidence in legal matters. For example,

359 The National Research Council, "Strengthening Forensic Science in the United States: A Path Forward", 2009, ISBN: 978-0-309-13130-8.

360 Reference Manual on Scientific Evidence - Second Edition - Federal Judicial Center, 2000.

361 F. Cohen, J. Lowrie, and C. Preston, “The State of the Science of Digital Evidence Examination”, Seventh IFIP WG 11.9 International Conference on Digital Forensics, 2011-01-30.

362 F. Cohen, “Update on the State of the Science of Digital Evidence Examination”, Conference on Digital Forensics, Security, and Law, May 29-31, 2012

If you cannot observe DFE without altering it, every time you examine it, it changes. How then can we state that what we present is what we originally got or examined?

If it is not trace evidence, how then did it come to be? In this case, it appears that the community members participating did not have common definitions.

If it is not finite granularity, how then can you present it as a finite set of 1/0 values (i.e., bits)?

A second approach in the 2010 study was a survey of the literature. In this survey, 125 reviews of 95 unique published articles (31% redundant reviews representing about 19% of the total corpus of peer reviewed papers in the field) was undertaken to identify the presence or absence of the underlying elements of a science (i.e., a common language for communication, concepts are defined, methodologies are defined by or used in publications, testability measures are defined by or tests described by the publication, and that validation methods are defined by or applied within the publication). Redundancy in reviews provides for assurance against reviewer bias, and redundancy showed only 9% of entries in which reviewers disagreed about the presence or absence of these indicators of scientific basis. Of these reviews,

- 88% have no identified common language defined,
- 82% have no identified scientific concepts or basis identified,
- 76% have no identified testability criteria or testing identified,
- 75% have no identified validation identified, and
- 59% identify a methodology.

Indications based on visual inspection of the time sequence of primary classifications suggest that methodology was an issue up to about 2001. At that point, evidence analysis, interpretation, and attribution became focal points. Then in 2005, methodology again became a focus. Then, in the middle of 2009, analysis started to again become more dominant.

These two preliminary studies individually suggested that (1) scientific consensus in DFE examination was lacking in the broad sense, but that different groups within that overall community may

have limited consensus around areas in which they have special expertise, and (2) the peer-reviewed publication process was not bringing the sorts of elements typically found in the advancement of a science toward such a consensus. Publication results also suggest that methodologies are a substantial focus of attention and that perhaps the most significant challenge may be in the development of a common language to describe the field. This is confirmed by the substantial portion of “don't know” responses to consensus surveys.

The follow-on study with data collected in 2011 was focussed on determining whether and to what extent the lack of consensus above random levels was due to the lack of common language and definitional issues or actual disagreement about technical issues. This study used 10 of the 11 technical questions from the earlier study essentially word for word, but added definitions of terms before each statement evaluated and asked respondents to identify agreement levels both to the definitions and to the statements based on the assumed definitions given.

The result was that agreement in excess of random expectations was present for 6 of the 10 statements and 8 of the 10 definitions. Consensus above random agreement was found for “Digital evidence is finite in granularity in both space and time.” (80%), “It is normally possible to observe digital information without altering it.” (73%), “It is normally possible to duplicate digital information without removing it.” (81%), “Digital evidence is normally latent in nature.” (86%), “Computational complexity limits digital forensic analysis.” (84%), and “The physics of digital information is different than that of the physical world.” (73%) With the exception of “Digital evidence is only sequences of bits.”, there did not appear to be substantial disagreement in the community, and this is a topic that is part of ongoing debate within the community, so this results is reasonably expected.

It appears that a significant source of lack of consensus in the 2010 study was related to the lack of common language and agreed upon terminology in the field also identified in that study. While these studies are not definitive, they certainly suggest a consensus surrounding many of the fundamentals of the field and confirm the lack of standards in publications, education, and training.

Tomorrow

This book is all about bringing a change. It makes an attempt at creating a basis for scientific examination, complete with a theory in the form of a physics and a model, methods for applying the theory, an experimental basis for testing hypotheses and theories, a system of measurement and process for carrying out such measurements, and an approach to adapting the theory at every level as experimental results are applied to hypotheses.

Throughout the book, the use of specific words in contexts are given as suggestions as to how to present results so that they are understood within the scientific community and won't be misunderstood outside of the community, once the words are more commonly used. While presentations may differ greatly, the careful use of words like “suggests”, “traces”, and “consistent” are intended to provide a basic notion of both what can and what cannot rightly be said about examination and its results.

The book has used and reused the principles of information physics, the limits of human cognition, and common fallacies to try to point out the limits of the science as we know it today, and to support the notion of how to present and how to challenge DFE and the examination process used to bring it to court. Again, while some may favor one side or view over others, this book has tried to be careful in presenting how far examiners can and should go, and when they step over the line, how to deal with them.

While more rigor will likely be applied over time and the basic outlines of information physics are slowly being filled in, the present effort represents only a snapshot of what is here today.

What lies ahead for this field? Will it turn into an eternal endeavor to build an ever advancing science that remains behind the state of the art in computing as it seems to have done for the last 50 years? Or will the application of science and advancement of theory bring it into a position of leading technology and providing an approach that will work for the foreseeable future? Will it be the DNA of the information age, or will it fall to the ages as another way for the rich to retain their power and enforce their will on those who cannot afford to fully litigate a complex case involving the massive volume of information potentially available?

It is clearly the view of this book that justice can only be found in the bright light of science and the scientific method and endeavor. Will the approach of this book end up the prevailing scientific theory in digital forensics? Nobody knows, and frankly, it's not important.

The important thing to understand is that science needs to be applied to digital forensic evidence examination, and this book is one attempt to lead the field down this path. Refute it, confirm it, dispute it, ... just don't burn it!

Questions

1. Given the state of consensus suggested by the referenced study, what sorts of challenges to examiners and their examinations are likely to result?
2. Given the lack of agreement over such fundamental issues as whether DFE can be examined without alteration, what levels of agreement or disagreement do you anticipated for higher level concepts, like the association of a user identity with an individual? Will there be more or less consensus? Or will we only find out by doing more studies?
3. Given the small corpus of peer reviewed publications in the field, and the apparent lack of scientific elements in those publications, how should the community change their approach to peer review and publication to further science?
4. Is DFE examination going to become a scientific endeavor with all of the rigor and complexity that this brings?
5. Are you going to help it move that direction or try to prevent it from going there?
6. How can the approach of this book help to move things forward, and how will it hold the field back?
7. What is your scientific theory of DFE examination, and how does it work? How does it differ from the one in this book? And when are you going to write your book that moves from this theory onto the next one?
8. Did you find any errors or mistakes in this book? Point them out. How else will science proceed?

Extended outline, references, and glossary

Extended outline

Extended outline

Thank you for reading this book all the way to the end!
Your comments and corrections are always welcomed

Glossary [and comments]

Traces, events, and records

Trace := (digital forensics) A set of bit sequences produced from the execution of a finite state machine.(FSM)

Structured trace := A trace that follows a particular defined pattern.

Unstructured trace := A trace that is not structured. [Typically image data such as from sound, vision, or other external sensors.]

Derived trace := A trace generated by the examiner from another trace.

Constructed trace := A trace constructed from a reconstruction process.

C-trace := Constructed trace.

Original trace := A trace produced from evidence in the matter.

O-trace := Original trace.

Complete trace := A trace containing all inputs, states, and outputs of a finite state machine (FSM).

Partial trace := A trace that is not a complete trace.

Incomplete trace := A partial trace from which a complete trace cannot be uniquely reconstructed.

Event := (forensics) A claimed, asserted, or stipulated state of affairs or act.

Anchor event := An event asserted by the examiner based on personal experience or other authority and that can be linked to the issues in the case. [e.g., A time stamp from an external mechanism that the examiner has personal knowledge of.]

Record := A document created (i.e., made or received and set aside for action or reference) in the course of activity as an instrument and by-product of it. [All digital records are traces, but not all traces are records]

Internal record := A record meant for transmission over time.

External record := A record record meant for transmission across space.

Legal record := A record whose existence in writing is required by the juridical and/or administrative system within which it is created.

Public record := A record issued by a public person. [see below]

Nonlegal record := A record whose written form is discretionary.

Supporting record := A record that helps to carry out activities in which it participates (e.g., a map, note, plan, presentation, etc.) [Does not provide evidence that any such act was actually carried out]

Narrative record := Free-form communications of information (e.g., memos, messages, etc.) [Is not adequate to show that any such act was actually carried out.]

Instructive record := A record that indicates the form in which something is to be presented or done (e.g., manuals, regulations, instructions for filling out forms, etc.)

Enabling record := Records that either (1) enable performance of a mechanism (e.g., firmware or an operating system), (2) execute business instructions (e.g., a workflow application), (3) conduct experiments (e.g., a control program for a robotic mechanism), or (4) data used in or produced by analysis or observation.

Original record := The first manifestation of a complete and effective record, either received or stored, depending on whether the record is external or internal. [This is essentially never available for DFE examination because of it's physical nature.]

Draft := A document prepared for the purpose of correction, and meant to be provisional and temporary.

Copy := A reproduction of another document. [The other document could be an “original”, “draft”, or another “copy”]

Copy in the form of original := A copy that is identical to the original in all respects, but produced at a later time. [This is a physical copy of the media, which is outside of the realm of digital forensics.]

Imitative copy := A reproduction of both the form and content of a record. [This is what is typically available and called an “exact”, “bit image”, or “forensically sound”, copy in digital forensics.]

Exact copy := (forensics) imitative copy.

Bit image copy := (forensics) imitative copy.

Forensically sound copy := (forensics) imitative copy.

Simple copy := A transcription of the record content. [The text]

Inserts := A copy of a record or part of it contained within another original record.

Medium := (diplomatics) The physical carrier of a record.

Form := (diplomatics) The rules governing the representation of an act in writing.

Archive := (diplomatics) Sedimentations of the natural documentary residue of activities.

Archives := The whole of the documents made or received in the course of activity and kept for action or reference. [In archives, there is one archive for

each physical or juridical person, or creator. Therefore, each archives (or archival fonds, the terms being synonyms) is a whole of the records made by one creator and their interrelationships.]

Archival bond := (diplomatics) The relationship of a record to the other records within the archives in which it exists.

Provenance := from the Latin "prōvenīre", which means "to come forth", (pro-, convene, -ant). Identification of the origins and path by which something came to be.

Procedures and processes

Procedure := (diplomatics) A formal sequence of steps by which a transaction is carried out.

Procedure := (forensics) A formal sequence of steps by which an examiner examines traces.

Transaction := an act aimed to create, modify, maintain, or extinguish relationships between two or more physical or corporate persons. [Some acts, especially transactions, occur in writing or other documentary forms, thereby resulting in records.]

Process := (diplomatics) is a series of motions by which a person carries out acts, including those acts involved in a procedure. [These are the physical acts undertaken]

Process := (computers) a sequence of programmed instructions and related data executing within an operating environment. [There is typically a process identification number within the operating system structures, and there may be “threads” of execution by which multiple execution streams are simultaneously available to execute]

Persons

Person := The subject of a right or duty. [They are recognized by the legal system as capable of acts.]

> **Physical person** := A human being.
>
> **Juridical person** := A corporation or similar legal entity.
>
> **Succession** := A position or title. [e.g., The President]
>
> **Public person** := A person with responsibility for the administration of matters regarding the people as a whole [i.e., A person authorized to issue a public record.]
>
> **Private person** := Any person not a public person.
>
> **Author** := The person with the competence (i.e., authority and capacity) to issue the record.

Writer := The person competent for the articulation and disclosure of the record.

Addressee := The person for whom the record is intended.

Creator := The person in whose archives a record exists.

Originator := The person responsible for the electronic account or space in which the record was generated or from which it is sent.

Mens rae := A guilty state of mind.

Examination and computers

Analysis := Methods used to determine consistency or inconsistency of traces and events. [Typically, trace typing, generating derived traces, making various comparisons, and other similar processes.]

Interpretation := A cognitive process used by the examiner to understand the nature of traces and events in context and associate them with issues at hand. [It may be thought of as associating meaning with traces and events.]

Attribution := An interpretation of causality. [Typically identifying plausible (cause effect) sequences consistent with available traces and events. Particularizing or individualizing traces to candidate causes.]

Reconstruction := An experiment testing hypothesized causal chains. [Used to demonstrate consistency or inconsistency with hypothesized sequences.]

Presentation := A method by which traces (i.e., latent evidence) are make into something that can be sensed and observed by humans.

Characteristic := Trace type, syntax, and structure.

Feature := Trace content [e.g., Sequences of words, types of spelling errors, etc.]

Symbol set := A mapping between bit sequences and symbols they represent in an alphabet.

Octet := An 8-bit sequence.

Byte := An 8-bit sequence at a defined boundary.

Trace type := The thing that a trace is intended to represent when generated.

Typing := (forensics) A process by which the type of a trace is hypothesized for examination. [Traces may be retyped after further examination based on consistency analysis.]

Particularization := A process by which a typed trace is associated with a specific use or source.

Individualization := A process by which a trace is associated with an single specific person, process, or mechanism.

Identifier := A trace placed in records intended to associate the trace with a particular person, process, or other thing.

Indicator := Traces and/or events often associated with or produced by other known traces, events, or mechanisms.

Equivalent content := (inexact matches) The same content in different format.

Normalization := Conversion into a common commensurable format.

Nominal metrics := Lists of things with no basis for formal comparison.

Ordinal metrics := Implies a partial ordering.

Interval metrics := Implies the ability to count things not against any standard.

Ratio metrics := Additive, comparable, and normalized to a common zero value.

Compensatory damages := Damages that cover actual injury or economic loss. Compensatory damages are intended to put the injured party in the position they were in prior to the injury. They are also called "actual damages."

> **Physical Damage** := There are physical damages to the computer system. This is almost never the case in DFE examination.
>
> **Conversion** := The computer system was no longer usable at all by its possessor. This rarely occurs as long as the possessor has physical control and can rebuild the system for some useful purpose. Some attacks can result in the need for physical repair, like replacement of the BIOS chip.
>
> **Deprivation** := The possessor was significantly deprived of use to the point where the basic function of the computer was obstructed or completely lost. This results from malicious attacks, when software fails, from configuration errors, and from many other causes.
>
> **Lost value** := The chattel lost some value, quality, or its physical condition was harmed, but this does not include the mere alteration of content where that does not deprive the possessor of use. This may include leakage of confidential information, alterations that cause the system or applications not to function, and many other similar things.
>
> **Lost rights** := The possessor was deprived of some other legally protected interest such as a copyright, patent, or other interest or right. Trade secrets disclosed might be an example of this.

Damages must be:

> **Quantified** := The damages must be reasonably quantified by measurements taken. This means that the examiner must be able to measure something from traces and events that allows the quantitative value of damages to be determined.
>
> **Time framed** := The possessor must be deprived for a substantial period of time. The examiner must be able to identify the time frame over which

the deprivation took place and it must be substantial relative to some standard.

Tangible := Damages must be the result of tangible trespasses and not merely the result of intangible ones, like electromagnetic emanations that do not deprive use. The examiner must be able to show that the trespass occurred based on traces and events that demonstrate effects on the chattels owned by the possessor.

Unmitigatable := The possessor must reasonably act to mitigate harm. The examiner should be able to show that diligent efforts were applied to mitigate the harm by examination and analysis of changes to the system.

Uninvited := The recipient must not invite the harm if they are going to claim damages. For example, if the harm comes from signing up to a service that is provided, the use of resources by the service is not actionable. Traces may be consistent or inconsistent with this assertion.

Causal := The damages must be proximately caused by the other party. The examiner must be able to show proximate causality at some level of certainty by consistency of causality with the traces and events.

Wording in reports

Suggests := imply as a possibility ("The [traces / events] suggests ...") - calls to mind - propose a hypothesis or possible explanation.

Indicates := a summary of a statement or statements or other content codified ("His statement indicates that ...") OR a defined set of "indicators" are present and have, through some predefined methodology been identified as such ("The presence of [...] (smoke) indicates [...] (fire)")

Demonstrate := exemplify - show - establish the validity of - provide evidence for ("The reconstruction demonstrates that ...")

Correlates := a statistical relation between two or more variables such that systematic changes in the value of one variable are accompanied by systematic changes in the other as shown by statistical studies ("Based on [statistical analysis method(s)], the use of the "KKJ" account is correlated (p=95%) with ...")

Match := an exact duplicate ("These two documents have matching publication dates, page counts, ...")

Similar := A correspondence or resemblance as defined by specified and measured quantities or qualities ("The 18 files were similar in that they all had syntax consistent with HTML, sizes under 1000 bytes, ...")

Relate := A defined and specified link ("The file system is related to FAT32 in that FAT32 was derived from ...")

Associate := Make a logical or causal connection with basis provided. ("I associate these bit sequences with program crashes because …")